



# natural

---

Version 4.1.2 for Mainframes | Release Notes

 **SOFTWARE AG**

This document applies to Natural Version 4.1.2 for Mainframes and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

© Copyright Software AG 1979 - 2003.  
All rights reserved.

The name Software AG and/or all Software AG product names are either trademarks or registered trademarks of Software AG. Other company and product names mentioned herein may be trademarks of their respective owners.

# Table of Contents

<b>Natural 4.1.2 Release Notes for Mainframes - Overview</b>	1
Natural 4.1.2 Release Notes for Mainframes - Overview	1
<b>Natural 4.1.2 Release Notes - General Information</b>	2
Natural 4.1.2 Release Notes - General Information	2
Introduction	3
Prerequisites	3
Operating/Teleprocessing Systems Required	3
Discontinued Support of Operating/Teleprocessing Systems	3
Assemblers Required	4
Documentation	5
Migration Hints	6
Natural IMS/TM Interface	6
Natural Roll Server	6
Natural Advanced Facilities	7
Using a Version 3.1 FUSER File to be Shared by Natural Versions 3.1 and 4.1	7
Application Programming Interfaces USR0340N and USR0341N	7
Compatibility	9
Applications Created with Previous Natural Versions	9
Execute Applications Cataloged with Natural Version 4.1 with Natural Version 3.1	9
Compatibility with Natural Version 5.1.1 for OpenVMS, UNIX and Windows	10
Compatibility with Natural Version 6.1.1 for UNIX and Windows	11
Natural System File Layout	11
Natural CICS Interface Macro Parameters	11
Natural IMS/TM Interface	12
Natural Com-plete/SMARTS Interface	12
Compiler	12
Arithmetic Operations	12
Natural Optimizer Compiler	12
Utility Activation	13
Application Programming Interfaces	13
Data Area Editor	13
SYSNCP Utility	13
Profile Parameters and Macros	14
Performance	14
Translation of System Library Output	15
Data View Definition	15
Position Calculation for INPUT, PRINT and WRITE Statement Corrected	16
LOGON Command	16
Special-Purpose Zaps	16
Dropped Functionality	18
NaturalX DCOM Functionality	18
ADASTAR Functionality	18
Natural CICS Interface Macro Parameters	18
Natural IMS/TM Interface Profile Parameters COLPSCR and LINPSCR	18
Natural Com-plete/SMARTS Interface Profile Parameters ADDBUF, CRELO, EDITWRK and NUCRELC	19
V22COMP Option Removed from COMPOPT Command	19
NTCPC Utility	19
SYSBUS System Command	19
SYSDB2 Catalog and Procedure Maintenance Support	19
Profile Parameters and Macros	19
Changes Introduced with Previous Natural Versions	20
Version Check at Session Initialization Refined	20

Version Check for Module NATCONFIG Introduced . . . . .	20
Length Calculation for Edit Masks with Leading Filler Character Corrected . . . . .	20
Natural Remote Procedure Call . . . . .	21
Determination of Window Size Corrected . . . . .	21
Changes to the Natural Load Library . . . . .	21
End of Maintenance of Natural Versions . . . . .	21
Natural and Other Software AG Products . . . . .	22
Information on Upcoming Releases . . . . .	24
Changes and Enhancements Planned for Natural Version 4.1.3 . . . . .	24
Changes and Enhancements Planned for Next Major Release of Natural . . . . .	24
Examples . . . . .	25
<b>Programming Language Enhancements . . . . .</b>	<b>26</b>
Programming Language Enhancements . . . . .	26
New Statements . . . . .	27
EXPAND Statement . . . . .	27
REDUCE Statement . . . . .	27
RESIZE Statement . . . . .	27
Enhanced Statements . . . . .	28
CALL Statement . . . . .	28
CALLNAT Statement . . . . .	28
CLOSE PRINTER Statement . . . . .	28
DEFINE DATA Statement . . . . .	28
DEFINE PRINTER Statement . . . . .	29
DEFINE WORK FILE Statement . . . . .	29
ESCAPE Statement . . . . .	29
FIND Statement . . . . .	30
HISTOGRAM Statement . . . . .	30
INPUT Statement . . . . .	31
PERFORM Statement . . . . .	32
READ Statement . . . . .	32
SEND METHOD Statement . . . . .	33
Size of Alphanumeric and Binary Variables . . . . .	33
Size of Data Elements . . . . .	33
Dynamic Variables . . . . .	34
Object Handles . . . . .	34
Optional Parameters . . . . .	35
SPECIFIED Option in Logical Condition . . . . .	36
MASK Option in Logical Condition . . . . .	36
System Variables Now Available under Natural for Mainframes . . . . .	37
New System Variables . . . . .	37
Changed System Variable . . . . .	38
More Precise Results for Floating Point Conversion . . . . .	38
Precision of Floating Point Format Results of an Arithmetic Operation Improved . . . . .	38
Arithmetic Operations with Date and Time Enhanced . . . . .	38
Evaluation of SQRT for (Un)Packed Number Enhanced . . . . .	38
Substring Evaluation in MASK Corrected . . . . .	39
New Compiler Options . . . . .	39
<b>System Commands, Editors and Utilities . . . . .</b>	<b>40</b>
System Commands, Editors and Utilities . . . . .	40
Enhanced System Commands . . . . .	41
CATALL Command . . . . .	41
COMPOPT Command . . . . .	41
DUMP Command . . . . .	42
LAST Command . . . . .	42
LIST Command . . . . .	42
STRUCT Command . . . . .	42

Removed System Commands . . . . .	42
COMPOPT Command . . . . .	42
SYSBUS . . . . .	42
New Utilities . . . . .	43
Natural Object Handler . . . . .	43
Enhanced Editors and Utilities . . . . .	44
Data Area Editor . . . . .	44
Map Editor . . . . .	45
NATLOAD/NATUNLD Utility . . . . .	46
SYSBPM Utility . . . . .	46
SYSDDM Utility . . . . .	46
SYSMAIN Utility . . . . .	47
SYSNCP Utility . . . . .	47
SYSPARM Utility . . . . .	47
SYSRDC Utility . . . . .	47
SYSTP Utility . . . . .	47
SYSWEB Utility . . . . .	47
Utility Activation . . . . .	48
Support of Activation without Logon . . . . .	48
Source Editing . . . . .	49
FREEGDA Parameter . . . . .	49
SYSNCP User Exit . . . . .	49
Application Programming Interfaces USR* in Library SYSEXT . . . . .	50
Use of USR* Programs . . . . .	50
Using a Version 3.1 FUSER File for Natural Version 3.1 and 4.1 . . . . .	50
Using a Version 3.1 FUSER File for Natural Version 4.1 only . . . . .	51
Using a New FUSER File for Natural Version 4.1 . . . . .	51
Migration . . . . .	51
New Application Programming Interfaces in Library SYSEXT . . . . .	51
Software AG Editor . . . . .	52
Debugger Enhancements . . . . .	52
Enhanced Display Functions . . . . .	52
Break Processing . . . . .	52
Commands . . . . .	52
<b>Database Interfaces . . . . .</b>	<b>53</b>
Database Interfaces . . . . .	53
Natural for Adabas . . . . .	54
Support of Adastar Discontinued . . . . .	54
Adabas-Related Natural Enhancements . . . . .	54
Natural for DB2 . . . . .	54
Natural for DL/I . . . . .	56
DL/I-Related Enhancements . . . . .	56
Note Concerning NDB or NSB Generation in LE370 Environment . . . . .	56
Natural for VSAM . . . . .	57
<b>Operating System and Teleprocessing Interfaces . . . . .</b>	<b>58</b>
Operating System and Teleprocessing Interfaces . . . . .	58
Natural for VSE/ESA . . . . .	59
Natural for BS2000/OSD . . . . .	59
New Keyword Parameters . . . . .	59
Changed Keyword Parameter . . . . .	59
Natural Com-plete/SMARTS Interface . . . . .	60
New Functionality . . . . .	60
Generation Parameter Changes . . . . .	60
Change in Installation . . . . .	61
Natural IMS/TM Interface . . . . .	62
New Functionality . . . . .	62

Error Codes Replaced by Natural Error Messages . . . . .	62
Natural CICS Interface . . . . .	64
New Functionality . . . . .	64
Generation Parameter Changes . . . . .	64
Change in Installation . . . . .	65
Natural CICS Interface and Natural UTM Interface . . . . .	66
New Keyword Subparameters for Macro NTSWPRM . . . . .	66
Support of New 64-Bit Architecture . . . . .	66
Improved Abend and Error Handling . . . . .	66
Natural Roll Server . . . . .	67
Authorized Services Manager . . . . .	67
SMARTS Print/Work File Support . . . . .	67
<b>Natural XML Toolkit . . . . .</b>	<b>68</b>
Natural XML Toolkit . . . . .	68
Purpose of the Natural XML Toolkit . . . . .	69
XML Toolkit on the Mainframe . . . . .	69
Natural XML Toolkit Functions . . . . .	69
Mapping Natural Data Definitions to DTDs . . . . .	70
Serializing the Data to XML . . . . .	70
Mapping DTDs to Natural Data Definitions . . . . .	70
Parsing the XML File and Assigning the Contents to a Natural Data Structure . . . . .	70
<b>Miscellaneous Changes and Enhancements . . . . .</b>	<b>71</b>
Miscellaneous Changes and Enhancements . . . . .	71
Profile Parameters . . . . .	72
New Profile Parameters . . . . .	72
New Parameter Macro . . . . .	73
Enhanced Profile Parameters . . . . .	73
Profile Parameters Dropped . . . . .	74
Retrieving Dynamic Profile Parameters . . . . .	74
Parameter Macro NTFILE . . . . .	75
Profile Parameter PLUGIN . . . . .	75
NATCONFIG Enhancements . . . . .	76
Natural Performance Enhancements . . . . .	76
Source-Change Inventory . . . . .	76
Delivery of Source Changes Enhanced . . . . .	76
Non-Sliding Year Window (YSLW Parameter) . . . . .	77
Stack . . . . .	77
Buffer and Storage Management . . . . .	78
Natural Global Buffer Pool . . . . .	78
NATSREX2/3 Example . . . . .	78
Reduction of Object Code . . . . .	79
Change/Enhancement Requests . . . . .	80
Natural Web Interface . . . . .	82
Internal Usage of Dynamic Variables and New Natural Features . . . . .	82
W3* Programs for Dynamic Variables and Dynamic Arrays of Dynamic Variables . . . . .	82
Use #HTTP_NEWLINE together with W3HTML . . . . .	83
Samples Library SYSWEB . . . . .	83
<b>New Features in NOC Version 4.1 . . . . .</b>	<b>84</b>
New Features in NOC Version 4.1 . . . . .	84
Code Generation and Performance Improvements . . . . .	85
Variable Caching . . . . .	85
EXAMINE . . . . .	85
RESET of Arrays . . . . .	85
Multiplication and Division of (Un)Packed Variables . . . . .	85
INDX Option with Parameter Arrays . . . . .	85
Objects with Copycode . . . . .	85

New Options . . . . .	86
Revised Options . . . . .	86
NODBG . . . . .	86
NOSGNTR=ON . . . . .	87
NOSGNTR=OFF . . . . .	87
LOOPS . . . . .	87
Incompatibilities . . . . .	88
NOSGNTR=ON . . . . .	88
NOSGNTR=OFF . . . . .	88
Conversion from Floating Point to Packed . . . . .	88
Multiplication and Division of (Un)Packed Variables . . . . .	88
RESET of Packed Numeric Variables . . . . .	88
MOVE Numeric to Numeric with same Precision and Length . . . . .	88
NOC 2.3 Zaps and Equivalent Option Settings . . . . .	89
NOCSTAT . . . . .	89
NOCSHOW . . . . .	89
Option Limit for NTOPT and OPT . . . . .	89
<b>Natural Security . . . . .</b>	<b>90</b>
Natural Security . . . . .	90
Environment Protection . . . . .	91
Logon Procedure . . . . .	91
Unsuccessful Logon Attempts . . . . .	91
Password Expiration Message . . . . .	91
Concurrent Modifications of a Security Profile . . . . .	92
User Profiles . . . . .	92
Time Differential and Time Zone . . . . .	92
Library Profiles . . . . .	93
Private Libraries . . . . .	93
Linking Administrators to an Unprotected Library . . . . .	93
Copying Libraries . . . . .	93
Copying, Renaming and Deleting Libraries . . . . .	93
Default Profiles . . . . .	94
Statement and Command Restrictions . . . . .	94
Utility Profiles . . . . .	95
SYSBPM . . . . .	95
Link Functions . . . . .	95
Functional Security . . . . .	95
Mailboxes in Batch Mode . . . . .	95
Interface Subprograms . . . . .	96
Direct Commands . . . . .	96
<b>Natural SAF Security . . . . .</b>	<b>97</b>
Natural SAF Security . . . . .	97
SAF Security Kernel . . . . .	97
SAF Online Services . . . . .	97
NSF Options . . . . .	97
Documentation . . . . .	97
<b>Natural Advanced Facilities . . . . .</b>	<b>98</b>
Natural Advanced Facilities . . . . .	98
Spool File . . . . .	99
New Functions . . . . .	99
CICS-Specific Enhancements . . . . .	99
IMS/TM-Specific Enhancements . . . . .	99
Spool File Options . . . . .	100
Server Options . . . . .	100
Report Display . . . . .	100
Printer Definitions . . . . .	100

CALLNAT Interfaces . . . . .	100
<b>Natural Connection Version 4.1 . . . . .</b>	<b>101</b>
Natural Connection Version 4.1 . . . . .	101
Support of Long and Dynamic Variables . . . . .	102
Work File Format and Attributes . . . . .	102
Installation Procedure . . . . .	102
Maximum File Transfer Record Length for Natural Connection . . . . .	102
<b>Natural RPC Version 5.1.1 . . . . .</b>	<b>103</b>
Natural RPC Version 5.1.1 . . . . .	103
New Features . . . . .	105
Support of Large and Dynamic Alpha and Binary Variables . . . . .	105
Maximum Length for Node and Server Names Increased to 32 Characters . . . . .	105
Support of the EntireX Broker ACI V6 . . . . .	105
Support of SSL for the TCP/IP Communication . . . . .	106
Support of EntireX Location Transparency . . . . .	106
New Application Programming Interface USR2035N . . . . .	107
New Application Programming Interface USR2073N . . . . .	107
New User Exit NATRPC99 . . . . .	108
Support of Natural Security Logon without Valid Password . . . . .	108
Enhanced Verification of the Natural Security User ID . . . . .	108
Prerequisites . . . . .	108
Enhanced Features . . . . .	109
Support of Multiple EntireX Broker Logons (USR2071N) . . . . .	109
Enhancements to Application Programming Interfaces . . . . .	109
Enhancements to Status Function RPCINFO . . . . .	109
SYSRPC Utility Enhancements and Modifications . . . . .	110
Generation of Service Directory (NATCLTGS) in User Library . . . . .	110
Support of Long Node and Server Names . . . . .	110
Support of EntireX Location Transparency . . . . .	110
New Properties for Local Directory . . . . .	110
Generation of Stubs in User Library . . . . .	110
Removal of 32K Restriction for Stubs . . . . .	111
Preserving Attribute Definitions of Existing Stubs . . . . .	111
Attribute Definitions as Comment in DEFINE DATA PARAMETER . . . . .	111
Support of Natural Security Utility Profiles . . . . .	111
Changed Features . . . . .	112
Implicit END TRANSACTION in a Conversation . . . . .	112
Release of Adabas Retain Sets . . . . .	112
Reworked Sample USR1071P . . . . .	112
Changes to RPCERR . . . . .	112
Changes to the Error Messages of the Server . . . . .	112
Changes to PIng and TErminate Messages . . . . .	113
Service Directory (NATCLTGS) in User Library . . . . .	113
Generation of Stubs in User Library . . . . .	113
Changes/Enhancements to Profile Parameter RPC . . . . .	113
RPCSIZE Considerations . . . . .	114
Unsupported Features . . . . .	114
Optional Parameters . . . . .	114



# Natural 4.1.2 Release Notes for Mainframes

## - Overview

These Release Notes inform you of the enhancements provided with Version 4.1.2 of Natural. This version contains all ZAPs, INPL updates, early warnings and source changes applied to Natural 3.1.6 as error corrections.

For background information, you should also read the Natural Version 2.3.3, 2.3.4, 3.1.2, 3.1.3, 3.1.4, 3.1.5 and 3.1.6 Release Notes for Mainframes which are available with Version 4.1.2 on the current Natural Documentation CD-ROM (in the folder RN-Archive) and on ServLine24: <http://servline24.softwareag.com>.

The following topics are covered:

- |  |   |
|--|---|
| ● General Information                            | Information on the documentation, availability, compatibility, prerequisites of Natural Version 4.1, other Software AG products with Natural 4.1, and examples. |
| ● Programming Language Enhancements              | Information on new and enhanced statements and system variables, as well as other enhancements to the Natural programming language.                             |
| ● System Commands, Editors and Utilities         | Information on new and enhanced Natural system commands, editors and utilities.   |
| ● Database Interfaces                            | Information on the Natural interfaces to Adabas, DB2, DL/I and VSAM.  |
| ● Operating System and Teleprocessing Interfaces | Information on the Natural interfaces to z/OS, VSE/ESA, BS2000/OSD and CICS, IMS/TM, Com-plete/SMARTS, UTM.   |
| ● Natural XML Toolkit                            | Information on Natural's support of XML processing.   |
| ● Miscellaneous Changes and Enhancements         | Information on all other changes, enhancements and new features.  |
| ● Natural Optimizer Compiler                     | Information on changes/enhancements to Natural Optimizer Compiler.  |
| ● Natural Security                               | Information on changes/enhancements to Natural Security.  |
| ● Natural SAF Security                           | Information on changes/enhancements to Natural SAF Security.  |
| ● Natural Advanced Facilities                    | Information on changes/enhancements to Natural Advanced Facilities.   |
| ● Natural Connection                             | Information on changes to Natural Connection.   |
| ● Natural Remote Procedure Call                  | Information on changes/enhancements to Natural RPC.   |

# Natural 4.1.2 Release Notes - General Information

The following topics are covered:

- Introduction
  - Prerequisites
  - Documentation
  - Migration Hints
  - Compatibility
  - Dropped Functionality
  - Changes Introduced with Previous Natural Versions
  - End of Maintenance of Natural Versions
  - Natural and Other Software AG Products
  - Information on Upcoming Releases
  - Examples
-

## Introduction

These Release Notes inform you of the enhancements provided with Version 4.1.2 of Natural.

Natural 4.1.2 contains all Zaps, INPL updates, early warnings and source changes applied to Natural 3.1.6 as error corrections.

## Prerequisites

- Operating/Teleprocessing Systems Required
- Discontinued Support of Operating/Teleprocessing Systems
- Assemblers Required

## Operating/Teleprocessing Systems Required

Natural Version 4.1 requires the following versions of the following operating/teleprocessing systems:

Product	Version
BS2000/OSD	3, 4 or 5
OS/390	2.10
z/OS	1.2, 1.3 or 1.4
VSE/ESA	2.6 or 2.7
z/VM	3.1, 4.2, 4.3 or 4.4
Com-plete	6.2.1 Patch Level 4 with SMARTS APS Version 2.7.2
CICS/TS	1.3 for OS/390, 1.1.1 for VSE/ESA
CICS/VSE	2.3
IMS/TM	7.1 or 8.1
UTM	4 or 5
TIAM	All versions available with OSD Version 3, 4 or 5.

Software AG provides Natural support for the operating/teleprocessing system versions supported by their respective manufacturers. Generally, when an operating/teleprocessing system provider stops supporting a version of an operating system, Software AG will stop supporting that operating system version.

Although it may be technically possible to run a new version of Natural on an old operating/teleprocessing system, Software AG cannot continue to support operating/teleprocessing system versions that are no longer supported by the system's provider.

## Discontinued Support of Operating/Teleprocessing Systems

Software AG will stop supporting Natural and all related add-on products on BS2000/OSD Version 3 on June 30, 2004.

## **Assemblers Required**

Natural Version 4.1 requires one of the following assemblers for the assembly of its source modules:

- "HL" Assembler Version 1.3 (IBM),
- "Assembh" Assembler (Siemens).

It may well be possible that the source modules can be assembled with older assemblers; however, Software AG cannot guarantee this.

## Documentation

With this release, a completely revised and updated set of documents is distributed on CD-ROM. In addition, the following changes and enhancements to the Natural documentation have been introduced with this release:

- **New Documentation Structure**

The structure of the online documentation and the navigation have been improved further. Also, a clearer distinction is made between documents concerning the Natural development environment and those dealing with the Natural programming language.

Please, note that the former General Information section of the Reference documentation has been reorganized and integrated in the Statements documentation (on the Natural Documentation Overview page, see Language > Statements > Statement Usage Related Topics).

- **Master Index for the Online Documentation**

Although the full-text search function delivered on the Natural documentation CD-ROM is very useful as a means of quickly locating all documents for a certain search term, it can also be frustrating in that the number of hits for a typical search is usually quite large and the order of the documents returned is arbitrary. To enable immediate access to frequently needed information on Natural key components such as statements, commands, parameters and utilities, an (as yet basic) master index function has been provided. This function can be found in the navigation bar, which is located in the top right-hand corner of each HTML page. For the next version, a considerable increase of the number of index tokens is planned.

- **Change in Terminology**

For consistency reasons, the documentation now distinguishes between "user exits" and "application programming interfaces". For the definitions of these terms, refer to the Natural Glossary on the Natural Documentation CD.

## Migration Hints

- Natural IMS/TM Interface
- Natural Roll Server
- Natural Version 4.1.1 Buffer Pool
- Natural Advanced Facilities
- Using a Version 3.1 FUSER File to be Shared by Natural Versions 3.1 and 4.1
- Application Programming Interfaces USR0340N and USR0341N

### Natural IMS/TM Interface

#### Use of the Non-Conversational Natural IMS/TM Interface

This information applies only for users of the non-conversational Natural IMS/TM interface.

If you want to use the Natural IMS/TM Interface Version 4.1 together with the Natural IMS/TM Interface Version 2.3 in the same IMS/TM MPP environment, you must use the Authorized Services Manager of Natural Version 4.1 for storing the simulated SPA. The following steps are required after you have installed base Natural Version 4.1:

1. Copy the module NATAU31B to the APF authorized library used by the Authorized Services Manager.
2. Start the Authorized Services Manager of Natural 4.1 for the subsystem identified by the Natural/IMS profile parameter SPATID.

The Authorized Services Manager will issue a message to indicate that the service is also established for a version prior to Natural Version 4.1.

Please keep in mind that all non-conversational Natural sessions in the same IMS/TM environment must use the same value for SPATID.

#### Use of Roll Files

This information only applies to users of Roll Files, that is, it applies to users who do not use a Roll Server.

If you want to share Roll Files between the Natural IMS/TM Interface Version 4.1 and the Natural IMS/TM Interface Version 2.3, the Roll Files must be formatted with the Roll File Initialization module NATRSFI of Natural Version 2.3.

Please note that the DD names of the Roll Files are identical for all versions of the Natural IMS/TM Interface. Therefore it is not possible to use Roll Files of different versions in the same MPP.

### Natural Roll Server

From z/OS Version 1.2 on, the Roll Server allocates its Local Roll Buffer in a Memory Object "above the bar". Use the MEMLIMIT parameter on the EXEC statement of the Roll Server started task to ensure that enough memory can be allocated "above the bar".

The allocatable amount is controlled by the MEMLIMIT parameter of the JCL EXEC statement and SMF exit IEFUSI. If not enough space for the LRB is available in memory objects, the Roll Server terminates with the message:

```
RSM0052 - Not enough storage for IARV64 GETSTOR - Increase MEMLIMIT
```

## Natural Version 4.1.1 Roll Server

If you want to migrate from an existing Natural Version 4.1.1 installation,

- stop all Natural Version 4.1.1 Roll Servers,
- re-format the Roll File using the Natural Version 4.1.2 Roll File formatting routine NATRSRFI,
- restart the Roll Servers,

before you start any Natural Version 4.1.2 session.

## Natural Advanced Facilities

A new spool file layout is provided for improved data storage and access performance. You can convert a Version 2.3 spool file to Version 4.1, but not a Version 2.2 spool file.

If you want to use a VSAM spool file with Natural for VSAM Version 4.1, you will have to create a new VSAM file cluster.

An existing spool file can be converted by executing the new format function. Old reports are not converted. If you are using a VSAM spool file, a new dataset for Detail Records must be assigned and initialized. The dataset for the VSAM Index file is no longer used.

## Using a Version 3.1 FUSER File to be Shared by Natural Versions 3.1 and 4.1

If you use an existing Natural Version 3.1 FUSER system file to be shared by Natural Versions 3.1 and 4.1, you must upgrade your Natural Version 3.1 installation to Version 3.1.6.

Natural Version 3.1.6 Service Pack I0010 or a subsequent Service Pack is required. Service Pack I0010 and all subsequent Service Packs contain all the necessary Version 3.1 based solutions for Natural Version 4.1.

This service pack is also required to provide certain system command functionality for source objects of type Function that may be created and executed with Natural Version 6.1.1 for UNIX and Windows. For details on the functionality provided, see Compatibility with Natural Version 6.1.1 for UNIX and Windows.

The V31COMP compiler option may be used to ascertain that Natural source objects that are edited and cataloged with Natural Version 4.1 can still be cataloged with Natural Version 3.1.



If you are using a Natural Development Server with Natural Version 5.1.1 on Windows, you cannot share a Version 3.1 FUSER file between Natural Versions 3.1 and 4.1. It is necessary that you upgrade your installation to Natural Version 6.1.1 before sharing the FUSER file. Natural Version 5.1.1 will not properly handle the features available with the above-mentioned Natural Version 3.1.6 Service Packs applied.

## Application Programming Interfaces USR0340N and USR0341N

The application programming interfaces USR0340N and USR0341N (Natural Buffer Pool Interfaces) delivered with Natural Version 3.1 will not work properly under Natural Version 4.1. If you are using copies within application libraries on the FUSER system file, you should replace the interfaces with the new Version 4.1 ones to run under Natural Version 4.1.

For further considerations to set up the environment for using application programming interfaces in different versions, refer to User Application Programming Interfaces USR\* in Library SYSEXT.



## Compatibility

- Applications Created with Previous Natural Versions
- Execute Applications Cataloged with Natural Version 4.1 with Natural Version 3.1
- Compatibility with Natural Version 5.1.1 for OpenVMS, UNIX and Windows
- Compatibility with Natural Version 6.1.1 for UNIX and Windows
- Natural System File Layout
- Natural CICS Interface Macro Parameters
- Natural IMS/TM Interface
- Natural Com-plete/SMARTS Interface
- Compiler
- Arithmetic Operations
- Natural Optimizer Compiler
- Utility Activation
- User Application Programming Interfaces
- Data Area Editor
- SYSNCP Utility
- Profile Parameters and Macros
- Performance
- Translation of System Library Output
- Data View Definition
- Position Calculation for INPUT, PRINT and WRITE Statement Corrected
- LOGON Command
- Special-Purpose Zaps

### Applications Created with Previous Natural Versions

Applications that were created with Natural Version 2.2, 2.3 or 3.1 for Mainframes can be executed with Natural Version 4.1 for Mainframes without any adjustments to the programs or any conversion or migration procedure being required. This applies also to programming objects that have been cataloged with the Natural Optimizer Compiler.



Applications cataloged with Natural Version 2.1 must be recataloged before execution with Version 4.1. This applies also to data areas that are to be used in programming objects.

Software AG strongly recommends that existing applications be recataloged with Natural Version 4.1 to take advantage of improved runtime handling.

### Execute Applications Cataloged with Natural Version 4.1 with Natural Version 3.1

To execute applications cataloged with Natural Version 4.1 with Natural Version 3.1, it is necessary to recatalog the application with Natural Version 3.1.

## Compatibility with Natural Version 5.1.1 for OpenVMS, UNIX and Windows

Natural Version 4.1 for Mainframes will be syntax-compatible with Natural Version 5.1.1 for OpenVMS, UNIX and Windows.

### Exceptions

Features available in Natural Version 5.1.1 for OpenVMS, UNIX and Windows, but not in Natural Version 4.1 for Mainframes:

- REQUEST DOCUMENT statement

Features available in Natural Version 4.1 for Mainframes, but not in Natural Version 5.1.1 for OpenVMS, UNIX and Windows:

- RESIZE statement
- MULTI-FETCH option for READ, FIND, HISTOGRAM statements
- ESCAPE MODULE and ESCAPE TOP REPOSITION statements
- TO keyword for READ and HISTOGRAM
- DYNAMIC SEQUENCE and WITH REPOSITION clauses for READ statement
- New comparators LESS/GREATER THAN and LESS/GREATER EQUAL for READ and HISTOGRAM statements
- Usage of dynamic variables in the statements DISPLAY, WRITE, PRINT, STACK, INPUT, REINPUT, IF/AT BREAK
- IF/AT BREAK statement with /n/ clause for binary variables
- MOVE SUBSTR statement for binary variables
- MOVE ALL statement for binary variables
- Advanced arithmetic operations (multiplication/division) with Date / Time
- COMPOPT system command options TQMARK, NMOVE22, MASKCME
- System variables \*LINE, \*TP, \*TPVERS
- MASK option JJJ
- Selection boxes (session parameter SB)

For further platform specific restrictions, see the respective documentation.

## Compatibility with Natural Version 6.1.1 for UNIX and Windows

Natural Version 6.1.1 for UNIX and Windows enables the creation and execution of Natural objects of type Function. Source objects of type Function can be stored in a Natural system file on mainframe computers, but execution of such objects is not possible. The restrictions are similar to those for source objects of type Dialog.

The following table lists the system command functionality that is supported for objects of type Function:

System Command	System Command Functionality Supported	Detailed Explanation
CATALOG CHECK STOW	No	Natural Error NAT0924 is issued.
DELETE	Yes	
EDIT	No	Natural Error NAT4439 is issued.
LIST	Yes	Subcommands CA (Catalog), ED (Edit) and ST (Stow) are not allowed.
LIST DIR	Yes	
READ	Yes	
RENAME	Yes	
SAVE	Yes	
SYSMAIN	Yes	
SYSOBJH SYSTRANS NATUNLD/NATLOAD	Yes	

## Natural System File Layout

The layout of the Natural system files has not been changed compared to Natural Version 3.1.

## Natural CICS Interface Macro Parameters

The parameter CICSPLX has been changed in the NCMDIR macro in NCISCPCB.

The parameter FLDLEN has been changed in the NCMPRM macro in NCIPARM.

For more information about the changes in the Natural CICS Interface, see Operating System and Teleprocessing Interfaces, Natural CICS Interface.

## Natural IMS/TM Interface

If in a multi-session environment the active session is terminated by a program-to-program switch, all suspended sessions are discarded.

In the Natural IMS/TM Interface, several error codes have been replaced by Natural error messages. For more information about the changes in the Natural IMS/TM Interface, see Operating System and Teleprocessing Interfaces, Natural IMS/TM Interface.

## Natural Com-plete/SMARTS Interface

In the Natural Com-plete/SMARTS Interface, several error codes have been replaced by Natural error messages.

The old macro NCMCFPRM has been replaced with a new macro NFMPRM.

For more information about the changes in the Natural Com-plete/SMARTS Interface, see Operating System and Teleprocessing Interfaces, Natural Com-plete/SMARTS Interface.

## Compiler

The V31COMP option has been provided for the COMPOPT command to reject syntax constructions that are supported by Version 4.1, but not by Version 3.1.

The NMOVE22 option has been provided for the COMPOPT command to replace special purpose zaps.

For more information about the changes in the Natural Optimizer Compiler, see System Commands, Editors and Utilities, COMPOPT Command.

## Arithmetic Operations

Format conversion from literals, alphanumeric fields or packed numeric fields to floating-point fields have been improved.

In an arithmetic result is of format F4, now fields of format F8 are used as intermediate result to improve the precision.

Performance and precision of the system function SQRT for packed and unpacked numbers has also improved.

These improvements may lead to slightly different results. However, these results will be of greater precision than those obtained with Version 3.1.

For more information, see Programming Language Enhancements, More Precise Results for Floating Point Conversion.

## Natural Optimizer Compiler

The options NODBG, NOSGNTR and LOOPS have been revised and now have a different effect. The modified effects of these parameters also make some special purpose zaps obsolete. Several Natural Optimizer Compiler algorithms have been adapted to the standard Natural arithmetic, thus guaranteeing identical program results regardless of whether the NOC is used for program generation or not.

For more information about the changes in the Natural Optimizer Compiler, see Natural Optimizer Compiler.

## Utility Activation

From Natural Version 4.1, Natural invokes a Natural utility without performing a logon to the corresponding utility library in the FNAT system file. As a result, the performance is improved and Natural preserves the global data area (GDA) and/or application-independent variables (AIV). The current user library and the steplib settings are maintained. The new behavior can have an impact on Natural Version 4.1 batch procedures that contain logon instructions to utility libraries.

For more information about the changes to the utility activation, see System Commands, Editors and Utilities, Utility Activation.

## Application Programming Interfaces

With Natural Version 4.1, the Application Programming Interfaces USR\* in Library SYSEXT will run in a special mode that removes the necessity to set further steplibs for processing. This will reduce the impact on the Natural buffer pool search logic and will improve performance significantly if user exits are used extensively within user written applications.

For more information about the changes concerning the Application Programming Interfaces, see System Commands, Editors and Utilities, Application Programming Interfaces.

## Data Area Editor

Data areas containing field definitions that make use of new features introduced with Natural Version 4.1 (e.g. dynamic variables) are stored in a new and extended source format in the FUSER system file. Such data areas cannot be used or edited with Natural Version 3.1.

Data areas containing only field definitions that are valid with Natural Version 3.1 are stored by default in the old format compatible with Natural Version 3.1 to allow for sharing the data area between a Natural Version 3.1 and Natural Version 4.1 environment.

The V31COMP compiler option may be used to ascertain that a data area that is edited and cataloged with the Natural Version 4.1 data area editor can still be cataloged with Natural Version 3.1.

For more information, see System Commands, Editors and Utilities, Data Area Editor.

## SYSNCP Utility

The user exits of the SYSNCP utility have been modified. For more information about the changes to the SYSNCP utility, see System Commands, Editors and Utilities, SYSNCP Utility.

## Profile Parameters and Macros

If one of the values 1, 2 or 3 has been specified for the ADAMODE parameter, but an Adabas subcomponent is unable to perform an Adabas X48 communication, an error message is issued.

For Natural Versions 3.1 and 4.1, the default parameter setting is ADAMODE=2. If you use SMA (System Maintenance Aid) to install Natural, the ADAMODE parameter is set to 0 (zero) in BS2000/OSD and Com-plete/SMARTS environments.

With Natural Version 4.1, you cannot specify a maximum size for the DATSIZE parameter any more, only a minimum size can be specified. Use the new profile parameter OVSIZE to limit memory allocation.

Because the Natural Turbo Plug-in is now an integral part of Natural, the only valid value for the PLUGIN parameter is now OFF.

The effect of the value 0 for the LT parameter (LT=0) has been changed, so that LT is consistent with other limits, such as MADIO and MAXCL.

The parameter macro NTFILE has been replaced by the new parameter macro NTLFILE with different syntax, but equivalent functionality.

The process of translating system library output as set by TS=ON has been changed. For more information, see Translation of System Library Output.

For more information about changes concerning profile parameters and macros, see Miscellaneous Changes and Enhancements, Profile Parameters.

## Performance

Several runtime algorithms for assignments, arithmetic operations and comparisons have been improved for better performance. In most cases, it is not required that the Natural objects are recataloged to benefit from the enhancements.

The Natural Turbo Plug-in is now an integral part of Natural. This includes the new buffer-pool search algorithm to reduce the time required to search for an object in the buffer pool and the buffer-pool cache to reduce system file accesses. In addition, the Adabas multi-fetch feature speeds up the process of loading objects into the buffer pool.

The compression rate has been enhanced to reduce both the compressed size of a Natural thread and the time needed for a roll-in/roll-out event (particularly at terminal I/O).

## Translation of System Library Output

With the profile parameter TS or the session parameter TS set to ON or the compilation option TSENABL=ON (set with the COMPOPT system command), output from Natural system libraries is translated using a translation table, which may be necessary for locations with non-standard lower-case usage.

With Natural Version 3.1, this translation is not performed for each field written into the Natural page buffer, but for the entire contents of the page buffer immediately before being sent to the screen. This may in some cases lead to incorrect results: when the contents of the page buffer comes from different programs with different TS/TSENABL settings - that is, some parts of the page are to be translated, others are not - the last applicable setting will be used, and the page will be either translated entirely or not at all. It may also occur that the last page is output after the end of the program when the TS=ON/TSENABL=ON setting is no longer available to Natural and this last page will then not be translated.

These translation errors will be corrected with Natural Version 4.1: the translation will be performed individually for each field at the time when it is written into the page buffer, according to the current TS/TSENABL setting that applies to each field. Consequently, the resulting output may not be the same as with Natural Version 3.1.

## Data View Definition

With Natural Version 3.1 for Mainframes, a data view may contain an ordinary group field with an index (or index range) definition, when periodic group fields follow. However, this is incorrect since the group field is no periodic group field as such. In addition, in Natural for UNIX or Windows this leads to a syntax error.

### Example:

Definition in DDM	DEFINE DATA View
P 1 AP PE-GROUP	1 V1 VIEW OF DDM..
G 2 AQ GROUP-IN-PE-GROUP	2 GROUP-IN-PE-GROUP (1:3)
2 AR PE-FLD1	3 PE-FLD1
2 AS PE-FLD2	3 PE-FLD2

With Natural Version 4.1. for Mainframes, this construct will also be rejected at compilation, and the source code of the affected programming objects must be changed.

The correction rule is just to replace the indexed group field by the corresponding periodic group name in the data view definition.

**Example:**

Definition in DDM	DEFINE DATA View
P 1 AP PE-GROUP	1 V1 VIEW OF DDM..
G 2 AQ GROUP-IN-PE-GROUP	2 PE-GROUP (1:3)
2 AR PE-FLD1	3 PE-FLD1
2 AS PE-FLD2	3 PE-FLD2

**Position Calculation for INPUT, PRINT and WRITE Statement Corrected**

With Natural Version 3.1, if the position of an output element in an INPUT, PRINT or WRITE statement is determined by both tabulation (*nT*) and space insertion (*nX*) and the tabulation position is greater than 1, for example WRITE 3T 4X #FIELD, the calculated position of the output item is by one byte less than the correct position.

With Natural Version 4.1, calculation of the position has been corrected. The calculation result is now the same as with Natural for UNIX or Windows. The new calculation applies if the programming object is recataloged. This change may lead to slightly different output.

This change does not affect output elements with

- only tabulation (*nT*) or
- only space insertion (*nX*) or
- both tabulation and space insertion with a tabulation position of 1, for example WRITE 1T 5X #FIELD.

**LOGON Command**

All parameters specific for the Natural Optimizer Compiler that have been set using the system command NOCOPT and that are not provided with initial default values will be reset when a LOGON command is issued.

**Special-Purpose Zaps**

As of Version 4.1, numerous special-purpose Zaps superseding the Version 3.1 special-purpose Zaps have been implemented as new parameters. The following table lists the non-sourced Version 3.1 Zaps and their Version 4.1 successors.

Version 3.1 Zap	Version 4.1 Zap
NA42001	NA61001
NA42003	NA61002
NA42102	NA61003 *
NA42112	NA61004
NA43009	NA61005
NA44094	NA61006
NA44107	NA61007
NI34007	NI61001
NV51002	NV61001
NV51001	NV61002



\* This Zap is an integral part of Natural Version 4.1.2 or above.

## Dropped Functionality

- NaturalX DCOM Functionality
- ADASTAR Functionality
- Natural CICS Interface Macro Parameters
- Natural IMS/TM Interface Profile Parameters COLPSCR and LINPSCR
- Natural Com-plete/SMARTS Interface Profile Parameters ADDBUF, CRELO, EDITWRK and NUCRELC
- V22COMP Option Removed from COMPOPT Command
- NTCPC Utility
- SYSBUS System Command
- SYSDB2 Catalog and Procedure Maintenance Support
- Profile Parameters and Macros

### NaturalX DCOM Functionality

As of Natural Version 4.1.2, the NaturalX DCOM functionality is no longer supported on mainframes. Therefore, the TYPE=DCOM option in the macro NTBPI has been removed.

The libraries SYSEXCOC and SYSEXCOM are no longer delivered with the Natural example INPL (EXPL).

The component-oriented language constructs of Natural (e.g. CREATE OBJECT, SEND METHOD) will however continue to be available for local execution.

### ADASTAR Functionality

As the ADASTAR functionality is no longer supported by the Adabas Version required for Natural 4.1, it is no longer supported by Natural Version 4.1.

### Natural CICS Interface Macro Parameters

The parameters CDATE, CTIME, ROLLFLS and SWPSIZE have been dropped from the NCMDIR macro in NCISPCB.

The parameters ASA, CDATE and CTIME have been dropped from the NCMPRM macro in NCIPARM.

For more information about the changes in the Natural CICS Interface, see the section Miscellaneous Changes and Enhancements, Natural CICS Interface.

### Natural IMS/TM Interface Profile Parameters COLPSCR and LINPSCR

The Natural IMS/TM interface profile parameters COLPSCR and LINPSCR are obsolete and have been dropped. Their functionality is now provided by the Natural profile parameter TMODEL.

If you have explicitly specified a value for COLPSCR and LINEPSCR, an error message will be issued during the compilation of the Natural/IMS interface parameter module and the generation of the parameter module will fail.

## Natural Com-plete/SMARTS Interface Profile Parameters ADDBUF, CRELO, EDITWRK and NUCRELC

The Natural Com-plete/SMARTS Interface profile parameters ADDBUF, CRELO and NUCRELC are obsolete and have been dropped.

Also the parameter EDITWRK is obsolete and has been dropped. The functionality of this parameter is now provided by the Natural profile parameter EDBP.

If you have explicitly specified a value for the above parameters, an error message will be issued during the compilation of the Natural/Com-plete interface parameter module and the generation of the parameter module will fail.

## V22COMP Option Removed from COMPOPT Command

With Natural Version 4.1.2, the V22COMP option of the system command COMPOPT (Allow old Version 2.2 Syntax) has been removed. Consequently, this option is also invalid for the NTCMPO macro, the CMPO profile parameter and the OPTIONS statement.

## NTCPC Utility

With Natural Version 4.1, the utility NTCPC (library SYSPC) is no longer supported. The delivery of the NTCPC utility is discontinued as of Natural Version 4.1.2. Software AG recommends that existing applications using the SYSPC modules be modified so that they can be used with the Natural Object Handler (SYSOBJH) instead.

## SYSBUS System Command

With Natural Version 4.1, the Natural system command SYSBUS is no longer available. Instead, you use the system command BUS which performs the same function.

## SYSDB2 Catalog and Procedure Maintenance Support

The Catalog and Procedure Maintenance functions are not adapted to DB2 Version 6. No further enhancements of these functions are planned for future versions. The procedure Maintenance does not run with DB2 Version 6 or higher. It is recommended to use the Create Procedure Statement or Predict instead.

## Profile Parameters and Macros

Because the NaturalX DCOM functionality has been dropped, the profile parameter DCOM, the macro NTDCOM and the value DCOM of the NTBPI macro keyword parameter TYPE have been dropped.

The IDSIZE and the WSIZE profile parameters have also been dropped.

For the Natural profile parameter RPC, the subparameter ACIPATT has been dropped because it has become obsolete since ACI is the only supported transport method.

The function MIGRATE for the parameter SFILE in the NVSPARM module has been dropped.

For more information about changes concerning profile parameters and macros, see the section Miscellaneous Changes and Enhancements, ProfileParameters.

## Changes Introduced with Previous Natural Versions

The following important changes that have been introduced with previous Natural Versions should be considered when you upgrade your installation to Natural Version 4.1:

- Version Check at Session Initialization Refined
- Version Check for Module NATCONFIG Introduced
- Length Calculation for Edit Masks with Leading Filler Character Corrected
- Natural Remote Procedure Call
- Determination of Window Size Corrected
- Changes to the Natural Load Library

### Version Check at Session Initialization Refined

With Natural Version 3.1.5, the check for matching versions of the Natural nucleus and the FNAT system file has been refined. To prevent unpredictable errors during session execution, any attempt to start a Natural Version 4.1 nucleus with a Natural Version 2.3 or 3.1 FNAT system file will be rejected.

### Version Check for Module NATCONFIG Introduced

With Natural Version 3.1.5, a check for matching versions of the Natural nucleus and configuration module NATCONFIG has been introduced. To prevent unpredictable errors during session execution, any attempt to start a Natural Version 4.1.2 (or above) nucleus with a NATCONFIG module from a previous system maintenance (SM) release or version will be rejected.

#### Note:

If you adapted an existing NATCONFIG module for your own purposes, you cannot continue using that module. Transfer your changes to the new NATCONFIG source before you assemble and link NATCONFIG as described in the corresponding installation description.

### Length Calculation for Edit Masks with Leading Filler Character Corrected

With Natural Version 3.1.5, leading filler characters specified in edit masks for numeric operands are no longer counted as being part of the edit mask. This may reduce the output length of numeric fields with an associated edit mask so that following output in the same line is shifted one position to the left. To apply the correction to an existing object, it must be cataloged with Natural Version 3.1.5 or above.

#### Example:

```
P(P3) = -12
DISPLAY P (EM=-*ZZZ)
```

Output before Natural Version 3.1.5:

```
  P
-----
-12
```

Output with Natural Version 3.1.5 or above:

```
  P
-----
-12
```

## Natural Remote Procedure Call

As the CSCI transport protocol is no longer supported, the CSCPATT keyword subparameter of the NTRPC macro is rejected with Natural Version 3.1.6. The CSCPATT subparameter of the RPC profile parameter is ignored.

The value CSCI is rejected if specified as transport protocol for the DFS, RDS or TRANSP keyword subparameters of the NTRPC macro or for the DFS, RDS or TRANSP subparameters of the RPC profile parameter.

## Determination of Window Size Corrected

According to the Natural Statements Manual, the size of a window defined by means of the DEFINE WINDOW SIZE AUTO statement is determined by the window data, and not by the size of the window title. With previous versions of Natural for Mainframes, the size of the window was determined by the size of the window title.

With Natural Version 3.1.6, this has been corrected to make Natural for Mainframes compliant with Natural on Windows and UNIX. Now the window title is truncated if its size exceeds the size of the window. The size of windows that have been defined using the SIZE AUTO clause may also be reduced.

## Changes to the Natural Load Library

With Natural Version 3.1.6, the following changes have been made to the Natural load library:

- The module PRDXR34 has been renamed to PRDXREF.
- The module SPENUC has been added to support Natural Construct Spectrum Version 4.4.1 or above.

## End of Maintenance of Natural Versions

- The maintenance for Natural Version 3.1.5 for Mainframes ends 6 months after the release of Natural Version 4.1.2.
- The maintenance for Natural Version 4.1.1 for Mainframes ends with the release of Natural Version 4.1.2.

## Natural and Other Software AG Products

To use the following Software AG products in conjunction with Natural Version 4.1, the following product versions (or above) are required:

Product	Prod. Code	Version
Adabas	ADA	7.1.3
Adabas Online System	AOS	7.1.3
Adabas SQL Server	AQA	5.1.2
Adabas Native SQL	SQL	2.2.1
Adabas Review	REV	4.2.2
Adabas Text Retrieval	TRS	2.1.4
Com-plete	COM	6.2.1 Patch Level 4 with APS Version 2.7.2
Con-form	CMF	3.4.1
Con-nect	CNT	3.4.1
Entire DB Engine	AER	1.5.5
EntireX Broker Stub	EXX	6.2
Entire Event Management	NCL	2.1.2
Entire Net-Work	WCP	5.8.1 (This product is required if you are using Natural Security in a heterogeneous environment.)
Entire Operations	NOP	4.1.2
Entire Output Management	NOM	2.2.1
Entire System Server	NPR	3.2.1
Entire Transaction Propagator	ETP	1.5.2 (ETP Version 1.5.2 does also run under Natural Version 3.1.6; for the Software AG products required with Natural Version 3.1.6, see the Natural Version 3.1.6 Release Notes in the Release Notes (RN) Archive on the current Natural Documentation CD).
Natural Advanced Facilities	NAF	4.1.2
Natural CICS Interface	NCI	4.1.2
Natural Com-plete/SMARTS Interface	NCF	4.1.2
Natural Connection	NTC	4.1.2
Natural Construct	CST	4.5.1
Natural Development Server	NDV	2.1.2
Natural for DB2	NDB	4.1.2

Product	Prod. Code	Version
Natural for DL/I	NDL	4.1.2
Natural for SQL/DS	NSQ	4.1.2 Note: IBM also refers to SQL/DS as DB2 Server for VSE & VM.
Natural for VSAM	NVS	4.1.2
Natural IMS/TM Interface	NII	4.1.2
Natural ISPF	ISP	2.5.2
Natural Optimizer Compiler	NOC	4.1.2
Natural Security for Mainframes	NSC	4.1.2
Natural SAF Security	NSF	4.1.2
Natural TIAM Interface	NRT	4.1.2
Natural TSO Interface	NTI	4.1.2
Natural UTM Interface	NUT	4.1.2
Natural VM/CMS Interface	NCM	4.1.2
NaturalX	NXX	No longer supported.
Predict	PRD	4.3.1
Predict Application Control	PAC	2.3.2 Service Pack 5
Predict Case	PCA	2.5.2 Service Pack 1
Review Natural Monitor	RNM	3.6.3
System Automatic Tools	SAT	3.1.2
Super Natural	NSN	3.3.3

Although it may be technically possible to run versions of other Software AG products older than the ones listed above in conjunction with a new version of Natural, Software AG cannot continue to support such combinations.

## Information on Upcoming Releases

- Changes and Enhancements Planned for Natural Version 4.1.3
- Changes and Enhancements Planned for Next Major Release of Natural

### Changes and Enhancements Planned for Natural Version 4.1.3

With the next system maintenance level of Natural (Version 4.1.3), the following changes and enhancements will be provided:

- Natural RPC Version 6.1
- Natural Com-plete/SMARTS Interface

#### Natural RPC Version 6.1

As of Natural RPC Version 6.1, the Natural server concept will be supported by the Natural RPC server. The benefits are in general:

- Conversations no longer block the Natural RPC server until they are closed. This means, a single Natural RPC server is able to process several concurrently opened conversations.
- The start of new replicates of a Natural RPC server is more efficient, because the overhead to initialize a Natural session is minimized.

Further benefits will be available under CICS:

- The remote CALLNAT can be executed under the user ID of the client
- A DB2 thread will no longer be blocked until the Natural DB2 server is terminated.

In addition, optional parameters will be supported in the parameter list of a remote CALLNAT execution. That is, the client may use the nX notation to indicate skipped parameters in its parameter list and/or may omit the rightmost skipped parameters. The remote CALLNAT on the server has to define the optional parameters with the keyword OPTIONAL and may check for their existence using the SPECIFIED condition.

#### Natural Com-plete/SMARTS Interface

The Natural Com-plete/SMARTS Interface will support the IBM LE/370 language environment.

### Changes and Enhancements Planned for Next Major Release of Natural

With the next major release of Natural (following Version 4.1), the following changes and enhancements will be provided:

- Discontinued Support of Applications Cataloged with Natural Version 2.2
- Discontinued Support of V31COMP Compiler Option
- Discontinued Support of Utilities SYSTRANS and NATUNLD/NATLOAD
- Changed Default of Data Area Source Format

#### Discontinued Support of Applications Cataloged with Natural Version 2.2

The next major release of Natural after Version 4.1 will require that all applications cataloged with Natural Version 2.2 be recataloged before execution with that version.

This will apply also to data areas that are to be used in programming objects.

Software AG strongly recommends that existing applications be recataloged with Natural Version 4.1 to take advantage of improved runtime handling.



### **Discontinued Support of V31COMP Compiler Option**

With the next major release of Natural after Version 4.1, the V31COMP compiler option will be dropped. This option can be used in Natural Version 4.1 to disallow the usage of new Natural Version 4.1 programming language enhancements for compatibility purposes with Natural Version 3.1.

### **Discontinued Support of Utilities SYSTRANS and NATUNLD/NATLOAD**

With the next major release of Natural after Version 4.1, the utilities SYSTRANS and NATUNLD/NATLOAD will cease to be available. The functionality provided by SYSTRANS and NATUNLD/NATLOAD is available with the Natural Object Handler which was introduced with Natural Version 4.1.

### **Changed Default of Data Area Source Format**

With Natural Version 4.1, the default format for storing data areas in the FUSER system file is the format compatible with Natural Version 3.1. With the next major release of Natural after Version 4.1, the default will be changed to the new and extended format introduced with Natural Version 4.1.

## **Examples**

The example library SYSEXV provides examples of the new features of Natural Version 4.1.

# Programming Language Enhancements

The following programming language enhancements are provided with Natural Version 4.1:

- New Statements
  - Enhanced Statements
  - Size of Alphanumeric and Binary Variables
  - Size of Data Elements
  - Dynamic Variables
  - Object Handles
  - Optional Parameters
  - SPECIFIED Option in Logical Condition
  - MASK Option in Logical Condition
  - System Variables Now Available under Natural for Mainframes
  - New System Variables
  - Changed System Variable
  - More Precise Results for Floating Point Conversion
  - Precision of Floating Point Format Results of an Arithmetic Operation Improved
  - Arithmetic Operations with Date and Time Enhanced
  - Evaluation of SQRT for (Un)Packed Number Enhanced
  - Substring Evaluation in MASK Corrected
  - New Compiler Options
-

## New Statements

The following new Natural statements are available with Natural Version 4.1:

- EXPAND
- REDUCE
- RESIZE

### EXPAND Statement

The new statement EXPAND is used to increase the size of the currently allocated storage of a dynamic variable.

In the statement, you specify the name of the variable and its desired size. If that size is smaller than the size of the storage currently allocated to that dynamic variable, the EXPAND statement has no effect.

For further information, see Dynamic Variables.

### REDUCE Statement

The new statement REDUCE is used to reduce the size of the used storage (available to the programmer).

In the statement, you specify the name of the variable and its desired size.

The storage allocated to the dynamic variable beyond the specified size may be released at any time, when the statement is executed or at a later time.

If the currently used size (as contained in the new system variable \*LENGTH) of the dynamic variable is greater than the given size, \*LENGTH is set to the specified size and the content of the variable is truncated (but not modified). If the specified size is larger than the size of the storage currently allocated to the dynamic variable, the REDUCE statement will be ignored.

For further information, see Dynamic Variables.

### RESIZE Statement

The new statement RESIZE is used to adjust the length of a dynamic variable to exactly the size specified.

- If the specified size is smaller than the used size (as indicated by \*LENGTH) of the dynamic variable, the used size is reduced accordingly.
- If the specified size is larger than the size of the storage currently allocated to the dynamic variable, the size of the storage allocated to the dynamic variable is increased. The currently used size (as indicated by \*LENGTH) of the dynamic variable is not affected and remains unchanged.
- If the specified size is the same as the size of the storage currently allocated to the dynamic variable, the execution of the RESIZE statement has no effect.

For further information, see Dynamic Variables.

## Enhanced Statements

The following Natural statements have been enhanced:

- CALL
- CALLNAT
- CLOSE PRINTER
- DEFINE DATA
- DEFINE PRINTER
- DEFINE WORK FILE
- ESCAPE
- FIND
- HISTOGRAM
- INPUT
- PERFORM
- READ
- SEND METHOD

### CALL Statement

The CALL statement provides the following enhancements:

- The limit of 32 KB for the maximum length per parameter has been removed.
- A new option, INTERFACE4, provides for enhanced parameter descriptions. Also, with this option, the number of parameters to be passed to the invoked non-Natural program (currently 40) has been raised to 32767.

### CALLNAT Statement

The CALLNAT statement provides the following enhancements:

- **Notation "nX"** - see Optional Parameters.
- **Parameter Transfer with Dynamic Variables** - see Dynamic Variables.

### CLOSE PRINTER Statement

The CLOSE PRINTER statement provides a new option that enables the hardcopy printer to be specified as Printer 0 to close the printer.

### DEFINE DATA Statement

The DEFINE DATA statement provides two new options to be specified in the *parameter-data-definition* of a DEFINE DATA PARAMETER statement:

<b>DYNAMIC</b>	If you define a parameter as DYNAMIC, its length will be determined at runtime. For further information, see Dynamic Variables.
<b>OPTIONAL</b>	By default, a parameter is defined without OPTIONAL, which means that a value <b>must</b> be passed from the invoking object to the parameter. If you define a parameter as OPTIONAL, a value can - but need not - be passed from the invoking object to this parameter. For further information, see Optional Parameters.

## DEFINE PRINTER Statement

The DEFINE PRINTER statement provides a new option that enables the hardcopy printer to be specified as Printer 0 to define a printer.

## DEFINE WORK FILE Statement

The DEFINE WORK FILE statement provides a new option TYPE UNFORMATTED allows you to specify that a work file is to be used in stream mode instead of the other new option FORMATTED for record-oriented files.

## ESCAPE Statement

The ESCAPE statement provides the following enhancements:

- ESCAPE TOP REPOSITION
- ESCAPE MODULE

### ESCAPE TOP REPOSITION

The new option ESCAPE TOP REPOSITION allows you to dynamically reposition within a READ statement loop that is being executed, and restart the READ loop with another start value.

When an ESCAPE TOP REPOSITION statement is executed, Natural immediately continues processing at the top of the active READ loop, using the current value of the search variable as a new start value.

At the same time, ESCAPE TOP REPOSITION resets the system variable \*COUNTER to "0".

ESCAPE TOP REPOSITION can be specified within a READ statement loop accessing an Adabas, DL/I or VSAM database. The READ statement concerned must contain the option WITH REPOSITION.

### ESCAPE MODULE

This new option ESCAPE MODULE allows you to stop an inline subroutine and continue processing with the programming object which has invoked the object containing the inline subroutine.

When used within a subroutine, the existing option ESCAPE ROUTINE causes processing to continue with the statement following the PERFORM statement that has invoked the subroutine. In the case of an **inline** subroutine, this would be within the same programming object. If nested subroutines are used, that is, if the PERFORM statement is itself contained within another inline subroutine, it would take a lot of coding to leave the active programming object entirely.

The new option ESCAPE MODULE, however, will not only stop the processing of the inline subroutine, but also of the programming object containing the inline subroutine; processing will then continue with the object invoking that programming object. This will be particularly useful when multiple nested inline subroutines are used, as a single ESCAPE MODULE statement will suffice to leave the programming object altogether.

ESCAPE MODULE is only relevant in **inline** subroutines. In external subroutines, subprograms and invoked programs, it has the same effect as ESCAPE ROUTINE.

As with ESCAPE ROUTINE, loop-end processing will be performed. If you specify the keyword IMMEDIATE, no loop-end processing is performed.

## **FIND Statement**

The FIND statement provides the following enhancement:

### **Multi-Fetch**

Traditionally, Natural retrieves database records one by one. However, Adabas's Multi-Fetch functionality makes it possible to retrieve more than one database record per database access.

To make use of this functionality, the FIND statement provides a new MULTI-FETCH option. With this option, you are able to specify the number of records to be retrieved per database access when the statement is executed.

The MULTI-FETCH option is available for accesses to Adabas databases only. For database updates, the MULTI-FETCH option cannot be used.

MULTI-FETCH only affects the way in which the records are retrieved from the database. The program's record-processing logic is not affected; that is, the number of FIND processing loops executed is the same as without MULTI-FETCH, and the records are still processed one by one.

## **HISTOGRAM Statement**

The HISTOGRAM statement provides the following enhancements:

- Dynamic Change of Reading Direction
- New Comparators
- Multi-Fetch
- End of Range Condition (ENDING AT) Controlled by Database

### **Dynamic Change of Reading Direction**

With Natural Version 3.1, the database field values to be retrieved by a HISTOGRAM statement could be read in ascending or descending sequence. This is determined by the keywords ASCENDING and DESCENDING in the SEQUENCE clause. Also, the VARIABLE option allows you to determine the reading direction at runtime. However, once the HISTOGRAM statement is executed, you cannot change the reading direction.

With Natural Version 4.1, the new keyword DYNAMIC is provided for the SEQUENCE clause: It allows you to change the reading direction from ascending to descending (or vice versa) within an active HISTOGRAM processing loop that is being executed, without having to restart the loop. After the keyword DYNAMIC, you specify a variable to which the value "A" (for "ascending") or "D" (for "descending") can be assigned. The DYNAMIC option is available for accesses to Adabas and DB2 databases.

### **New Comparators**

In addition to the comparators EQUAL TO, STARTING FROM and ENDING AT, Natural Version 4.1 provides the possibility to specify start/end values with the following options:

- LESS THAN
- GREATER THAN
- LESS EQUAL
- GREATER EQUAL

These new comparators are available for accesses to Adabas, DB2, DL/I and VSAM databases.

## Multi-Fetch

Traditionally, Natural retrieves database records one by one. However, Adabas's Multi-Fetch functionality makes it possible to retrieve more than one database record per database access.

To make use of this functionality, the HISTOGRAM statement provides a new MULTI-FETCH option. With this option, you are able to specify the number of records to be retrieved per database access when the statement is executed.

The MULTI-FETCH option is available for accesses to Adabas databases only.

MULTI-FETCH only affects the way in which the records are retrieved from the database. The program's record-processing logic is not affected; that is, the number of HISTOGRAM processing loops executed is the same as without MULTI-FETCH, and the records are still processed one by one.

## End of Range Condition (ENDING AT) Controlled by Database

With Natural Version 3.1, if the ENDING AT clause was used to limit the range of values to be read, Natural internally read one value beyond the specified ENDING AT value in order to determine the end of the range to be read. This was necessary due to restrictions inherent in the underlying databases.

With Natural Version 4.1, these restrictions no longer apply, and the ENDING AT value can now be determined by the accessed databases themselves. This means that Natural is able to read the values only until including the specified ENDING AT value, but not beyond.

As this may lead to different results and so as not confuse the "old" end-value mechanism with the "new" one, a new keyword, TO, is provided for the specification of the database-controlled end value. The existing ENDING AT clause is not affected and continues to yield the same results as before.

The new keyword TO is available for Adabas, DB2, DL/I and VSAM databases.

## INPUT Statement

The INPUT statement provides the following enhancements:

- Selection Boxes
- Edit Mask Free Mode

### Selection Boxes

Natural Version 4.1 provides the possibility to attach selection boxes to input fields. These selection boxes are similar to those used in graphical user interfaces and are a comfortable alternative to help routines attached to fields.

To assign a selection box to a field, the INPUT statement provides the new field attribute SB. With SB, you specify the contents of the selection box, that is, the values, or the name of an array field that provides the values, to be displayed within the selection box. The size and position of the selection box are determined automatically (using the same algorithm as for help windows).

For a field for which the field attribute SB is specified, a selection-box indicator "V" is displayed next to the field. To invoke the selection box, the user positions the cursor on the "V" and presses the help key. The selection box is then displayed as a window on the screen. If the list of values within the selection box is longer than the selection box itself, the user can scroll by placing the cursor on the "More/Top/Bottom" lines of the selection box and pressing ENTER. To select a value from the selection box, the user positions the cursor on the desired value and presses ENTER. The selected value is then copied into the input field.

The field attribute SB is only available for alphanumeric fields.

## Edit Mask Free Mode

The edit mask free mode is an alternative INPUT mode for entering numeric fields with an edit mask use. When activated (either at session startup with the new profile parameter EMFM or in a running Natural session via the terminal command %FM+), all or some of the edit mask insert characters may be left out from input.

## PERFORM Statement

The PERFORM statement provides the following enhancements:

- **Notation "nX"** - see Optional Parameters.
- **Parameter Transfer with Dynamic Variables** - see Dynamic Variables.

## READ Statement

The READ statement provides the following enhancements:

- Dynamic Change of Reading Direction
- New Comparators
- Multi-Fetch
- End of Range Condition (ENDING AT) Controlled by Database
- WITH REPOSITION for Non-VSAM Databases

### Dynamic Change of Reading Direction

With Natural Version 3.1, the records to be retrieved by a READ statement could be read in ascending or descending sequence. This is determined by the keywords ASCENDING and DESCENDING in the **sequence/range-specification**. Also, the VARIABLE option allows you to determine the reading direction at runtime. However, once the READ statement is executed, you cannot change the reading direction.

With Natural Version 4.1, the new keyword DYNAMIC is provided for the **sequence/range-specification**: It enables you to change the reading direction from ascending to descending (or vice versa) within an active READ processing loop that is being executed, without having to restart the loop. After the keyword DYNAMIC, you specify a variable to which the value "A" (for "ascending") or "D" (for "descending") can be assigned. The DYNAMIC option is available for accesses to Adabas and DB2 databases.

### New Comparators

In addition to the field/value comparators EQUAL TO, STARTING FROM and ENDING AT, Natural Version 4.1 provides the possibility to specify start/end values with the following options:

- LESS THAN
- GREATER THAN
- LESS EQUAL
- GREATER EQUAL

These new comparators are available for accesses to Adabas, DB2, DL/I and VSAM databases.

### Multi-Fetch

Traditionally, Natural retrieves database records one by one. However, Adabas's Multi-Fetch functionality makes it possible to retrieve more than one database record per database access. To make use of this functionality, the READ statement provides a new MULTI-FETCH option. With this option, you are able to specify the number of records to be retrieved per database access when the statement is executed. The



MULTI-FETCH option is available for accesses to Adabas databases only. For database updates, the MULTI-FETCH option cannot be used.

MULTI-FETCH only affects the way in which the records are retrieved from the database. The program's record-processing logic is not affected; that is, the number of READ processing loops executed is the same as without MULTI-FETCH, and the records are still processed one by one.

### **End of Range Condition (ENDING AT) Controlled by Database**

With Natural Version 3.1, if the ENDING AT clause was used to limit the range of values to be read, Natural internally read one value beyond the specified ENDING AT value in order to determine the end of the range to be read. This was necessary due to restrictions inherent in the underlying databases.

With Natural Version 4.1, these restrictions no longer apply, and the ENDING AT value can now be determined by the accessed databases themselves. This means that Natural is able to read the values only until including the specified ENDING AT value, but not beyond.

As this may lead to different results and so as not confuse the "old" end-value mechanism with the "new" one, a new keyword, TO, is provided for the specification of the database-controlled end value. The existing ENDING AT clause is not affected and continues to yield the same results as before.

The new keyword TO is available for Adabas, DB2, DL/I and VSAM databases.

### **WITH REPOSITION for Non-VSAM Databases**

Due to the introduction of the new ESCAPE statement option TOP REPOSITION, the WITH REPOSITION option of the READ statement is no longer restricted to VSAM databases, but is also available for Adabas and DL/I databases.

### **SEND METHOD Statement**

The SEND METHOD statement provides the following enhancement:

- **Notation "nX"** - see Optional Parameters.

## **Size of Alphanumeric and Binary Variables**

With Natural Version 4.1, the maximum possible size of an alphanumeric variable (Format A) has been increased from 253 bytes to 1 GB. The maximum possible size of a binary variable (Format B) has been increased from 126 bytes to 1 GB.

## **Size of Data Elements**

With Natural Version 4.1, the maximum possible size of a single data element (array or indexed group) has been increased from 32 KB to 1 GB.

## Dynamic Variables

In addition to removing the size limitations for alphanumeric and binary variables (see Size of Alphanumeric and Binary Variables), Natural Version 4.1 makes it possible to allocate the length of such variables dynamically at runtime.

As the maximum size of large data structures (for example, pictures, sounds, videos) may not be known exactly at the time an application is developed, Natural provides for the definition of alphanumeric and binary variables with the attribute DYNAMIC. The value space of variables which are defined with this attribute is extended dynamically at runtime when it becomes necessary (for example, during an assignment operation: `#picture1 := #picture2`). This means that large binary and alphanumeric data structures may be processed in Natural without the programmer having to define a length at development time.

The new Natural system variable `*LENGTH` is provided to obtain the value space (number of bytes) currently used and available to the programmer for a given dynamic variable at runtime.

For performance optimization and also to avoid problems with too much or too little allocated memory space, the new statements EXPAND, REDUCE and RESIZE have been introduced. If the space allocated for a dynamic variable is no longer needed, the REDUCE or the RESIZE statement can be used to reduce or to resize the allocated space (to zero or any other desired size). If the upper limit of memory usage is known for a specific dynamic variable, the EXPAND statement can be used to set the space used for the dynamic variable to this specific size.

Dynamic variables can be used, for example, in CALLNAT or PERFORM statements. It is also possible to define and use arrays of dynamic variables.

See also Large and Dynamic Variables and Fields in the Natural Statements documentation.

## Object Handles

With Natural Version 4.1, it is possible to define object handles within a global data area or as application-independent variables (AIVs).

## Optional Parameters

Natural Version 4.1 supports the use of optional parameters in subprograms, external subroutines and dialogs. Optional parameters may be used to expand an existing subprogram (for example, to provide additional parameters) without having to change all objects that use this subprogram.

An optional parameter is a field defined with the keyword **OPTIONAL** in the **DEFINE DATA PARAMETER** statement of an invoked object (subprogram, external subroutine or dialog). To such a field, a value can - but need not - be passed from an invoking object.

In the invoking statement (**CALLNAT**, **PERFORM** or **SEND METHOD**), the notation *nX* is used to indicate optional parameters for which no values are passed. With *nX* you specify that the next *n* parameters are to be skipped; that is, for the next *n* parameters no values are passed to the invoked object.

### Example:

Subprogram:	Invoking Object:
<pre> DEFINE DATA PARAMETER 1 #P1 (A10) 1 #P2 (A10) <b>OPTIONAL</b> 1 #P3 (A10) 1 #P4 (A10) <b>OPTIONAL</b> 1 #P5 (A10) <b>OPTIONAL</b> END-DEFINE ... </pre>	<pre> CALLNAT 'MY-SUB' #A #B #C #D #E or CALLNAT 'MY-SUB' #A <b>1X</b> #C <b>2X</b> or CALLNAT 'MY-SUB' #A #B #C <b>1X</b> #E </pre>

To check in the invoked object whether or not an optional parameter has received a value from the invoking object, the new **SPECIFIED** option to be used in a logical condition is available.

## SPECIFIED Option in Logical Condition

With the new SPECIFIED option to be specified in a logical condition, you are able to check whether or not an optional parameter in an invoked object (subprogram, external subroutine or dialog) has received a value from the invoking object.

If you process an optional parameter which has not received a value, this will cause a runtime error. To avoid such an error, you use the SPECIFIED option in the invoked object to check whether an optional parameter has received a value or not, and then only process it if it has.

### Example:

```
IF #OPTFIELD1 SPECIFIED THEN ... ELSE ...  
IF #OPTFIELD2 NOT SPECIFIED THEN ... ELSE ...
```

For a field not defined as OPTIONAL, the SPECIFIED condition will always be "TRUE".

## MASK Option in Logical Condition

With Natural Version 4.1, it is possible to check positions of a field for a date in Julian format. This will be particularly useful when a MASK option is used in conjunction with a MOVE EDITED statement that uses a Julian date in its edit mask.

See also the COMPOPT system command for enhancements related to the MASK option.

## System Variables Now Available under Natural for Mainframes

The following Natural system variables that have been available under Natural for Windows and UNIX are now also provided under Natural Version 4.1:

System Variable	Format	Default DISPLAY Header	Description of Contents
*CPU-TIME	I4	CPU-TIME	CPU time currently used by the Natural process in units of 10 ms. In environments where this variable is not supported, it contains the value 0.  For details, refer to the Natural Variables documentation.
*DATV	A11	DATV	Current date in the format <i>dd-mon-yyyy</i> (where <i>mon</i> is the leading three bytes of the month's name as in *DATG).
*DATVS	A9	DATVS	Current date in the format <i>ddmonyyyy</i> (similar to *DATV).
*HOSTNAME	A64	HOSTNAME	Name of the machine on which Natural is running.
*LENGTH( <i>field</i> )	I4	LENGTH	Currently used length (in bytes) of a <b>field</b> defined as a dynamic variable. See also Dynamic Variables.
*NATVERS	A8	NATVERS	Natural version excluding patch level information in the format <i>rr.vv.ss</i> , where <i>r</i> =release, <i>v</i> =version, <i>s</i> =system maintenance level (example: 04.01.01)
*PARAM-USER	A253	PARAM-USER	Name of the parameter module currently in use (contains blanks if PARAM= <i>name</i> has not been specified as a dynamic parameter).
*PATCH-LEVEL	A8	PATCH-LEVEL	Current Natural patch-level number as a string.
*PID	A32	PID	Unique identifier for the Natural session.

## New System Variables

The following Natural system variables are new as of Natural Version 4.1:

System Variable	Format	Default DISPLAY Header	Description
*LINE	I4	LINE	Contains the number of the line currently executed in a Natural object.
*TP	A8	TP	Contains the name of the TP subsystem under which Natural is running. This value is supplied by the operating system and may be subject to change.
*TPVERS	A8	TPVERS	Contains the version of the TP subsystem under which Natural is running. This value is supplied by the operating system and may be subject to change.

## Changed System Variable

### System Variable \*LANGUAGE

The value "0" of the system variable \*LANGUAGE which was tolerated in earlier Natural for Mainframes versions has been dropped to ensure compatibility with Natural for UNIX and Windows.

## More Precise Results for Floating Point Conversion

The format conversion for the transfer of data from literals (as in assignments, INIT or CONST clauses), alphanumeric fields (as in the VAL system function) or packed numeric fields to floating-point fields has been improved. This may in some cases lead to different results. However, these results will be of a greater precision than with Natural Version 3.1.

### Example:

`F(F8) = 5.4E-79`

Result with Version 3.1: +5.399999999999999E-79

Result with Version 4.1: +5.400000000000000E-79

## Precision of Floating Point Format Results of an Arithmetic Operation Improved

For arithmetic operations with a result of format/length F4 in previous Natural versions, format/length F8 is now being used to improve the precision of the results.

## Arithmetic Operations with Date and Time Enhanced

Multiplication and division are now allowed on intermediate results of additions and subtractions of formats D (date) and T (time).

## Evaluation of SQRT for (Un)Packed Number Enhanced

The evaluation of the mathematical system function SQRT for packed and unpacked numbers has been enhanced to improve performance and precision of the results.

## Substring Evaluation in MASK Corrected

The evaluation of a SUBSTRING clause in the MASK option has been corrected. This may lead to different results if the evaluated substring is shorter than the specified mask.

### Example:

```
IF SUBSTR(A10, 2, 3) = MASK(....) THEN
```

The length of the evaluated substring is 3, but the mask has length 4.

Result with Version 3.1: TRUE

Result with Version 4.1: FALSE

## New Compiler Options

The following new compiler options have been introduced:

MASKCME	The check for the mask character YYYY can be made compatible with MOVE EDITED.
NMOVE22	Assignment of numeric variables of same length and precision may be performed as with Natural Version 2.2.
TQMARK	The check for the translation of quotation mark.
V31COMP	Disable new syntax of Natural Version 4.1.

For further information, refer to system command COMPOPT.

# System Commands, Editors and Utilities

The following topics are covered:

- Enhanced System Commands
  - Removed System Commands
  - New Utilities
  - Enhanced Editors and Utilities
  - Utility Activation
  - Application Programming Interfaces USR\* in Library SYSEXT
  - New Application Programming Interfaces in Library SYSEXT
  - Software AG Editor
  - Debugger Enhancements
-



## Enhanced System Commands

The following Natural system commands have been enhanced for Natural Version 4.1:

- CATALL
- COMPOPT
- DUMP
- LAST
- LIST
- STRUCT

### CATALL Command

The CATALL command provides the following enhancements:

- The error list provided by the CATALL command has been expanded to include additional information on the causes of CATALL processing errors.
- A user exit has been provided which allows you to change the default settings of the main CATALL screen.

### COMPOPT Command

The COMPOPT command provides the following enhancements:

- With the MASK option in a logical condition, a valid year value must be in the range 0000 - 2699; with the MOVE EDITED statement, a valid year value must be in the range 1582 - 2699. Consequently, a year value found valid by a MASK option might lead to an error in a subsequent MOVE EDITED statement. To avoid this situation, the COMPOPT command provides a new option, MASKCME, which allows you to determine whether the year range to be considered valid by the MASK option is to begin with 0000 or 1582.
- With Natural Version 2.3, the internal handling of assignments between two Format N variables of the **same** length was changed, so as to be consistent with the internal handling of assignments between Format N variables of different lengths. One side effect of that change was that when these variables were redefined, this could in some cases lead to different results.  
With Version 4.1, the COMPOPT command provides a new option, NMOVE22, which allows you to determine whether the internal handling of assignments between Format N variables of the same length is to be inconsistent as in Version 2.2 or consistent as in all subsequent versions (the latter will be the default). The NMOVE22 option replaces the corresponding special-purpose Zaps that were supplied for Natural Versions 2.3 and 3.1. To provide upward compatibility at runtime for objects cataloged with previous Natural Versions **without** the Natural Optimizer Compiler, a Zap will be provided with Natural Version 4.1. This Zap will not be provided for subsequent Natural versions. Objects cataloged with previous Natural Versions with the Natural Optimizer Compiler are not affected and will continue to work as before. Affected objects should be recataloged as soon as possible with Natural Version 4.1, with NMOVE22=ON being set.
- The option V31COMP allows to reject syntax constructions that are supported by Version 4.1, but not by Version 3.1. This option will be available only with Natural Version 4.1 to allow a smooth transition. It will be removed again with the next major release of Natural after Version 4.1.
- The option TQMARK to check for the translation of quotation marks.
- With the option PCHECK set to ON, the compiler checks the number, format, length and array index bounds of the parameters that are specified in a CATALL statement. Also, the OPTIONAL feature of the DEFINE DATA PARAMETER statement is considered in the parameter check.

**Note:**

See also Removed System Commands, for information on the V22COMP option of the system command COMPOPT.

## DUMP Command

The DUMP command provides the following enhancements:

- An SRCE function to show the inventory of source changes applied per product (similar to the ZAPS function).
- An SVAR function to show TP monitor or operating system dependent system variables and additional information.

## LAST Command

The LAST command provides the following enhancement:

In previous Natural versions, "LAST \*" displayed a window showing the last 9 system commands that were issued. With Natural 4.1, this window shows the last 20 system commands.

## LIST Command

The LIST command provides the following enhancement:

- When the LIST command displays a list of objects, these objects are in alphabetical order of their names. With Natural 4.1, it is also possible to sort the list of objects by type, date, etc.

## STRUCT Command

The STRUCT command now supports two-digit field levels.

## Removed System Commands

The following Natural system command or command option was dropped as of Natural Version 4.1:

### COMPOPT Command

The V22COMP option of the system command COMPOPT (Allow old Version 2.2 Syntax) has been removed. Consequently, this option will also be invalid for the NTCMPO macro, the CMPO profile parameter and the OPTIONS statement.

### SYSBUS

The Natural system command SYSBUS is no longer available. Instead, you use the system command BUS which performs the same function.

## New Utilities

The following new Natural utility is provided with Natural Version 4.1:

### Natural Object Handler

The Natural Object Handler processes objects for the purpose of application distribution. This utility is invoked with the system command SYSOBJH and combines the functionality currently provided by the utilities SYSTRANS and NATUNLD/NATLOAD. The utilities SYSTRANS and NATUNLD/NATLOAD will cease to be available with the next major release of Natural (following Version 4.1).

The Natural Object Handler enables you to

- unload objects in the source environment to work files, and then load these objects from work files into the target environment.
- process Natural programming objects, resources, DDMs, error messages, Natural-related objects, Natural command processors, external objects, and Adabas FDTs.
- perform unloading and loading in an internal format (as with the NATUNLD/NATLOAD utility) or in a transfer format (as with the SYSTRANS utility).
- process work files that were created with the utilities SYSTRANS and NATUNLD/NATLOAD.

Work files created with the Natural Object Handler in transfer format can be processed by the utility SYSTRANS on all platforms.

## Enhanced Editors and Utilities

The following enhancements are provided with Natural 4.1:

- Data Area Editor
- Map Editor
- NATLOAD/NATUNLD
- SYSBPM
- SYSDDM
- SYSMAIN
- SYSNCP
- SYSPARM
- SYSRDC
- SYSTP
- SYSWEB

### Data Area Editor

The Natural data area editor provides full support of the following enhancements of the Natural language:

- Alphanumeric and binary variables with a length of up to 1 GB  
(See also Miscellaneous Changes and Enhancements, Size of Alphanumeric and Binary Variables.)
- Dynamic variables  
(See also Programming Language Enhancements, Dynamic Variables.)
- Array indices of up to 10 digits; the maximum index value is 1073741824, the maximum size of an array may now be 1 GB  
(See also Miscellaneous Changes and Enhancements, Size of Data Elements.)
- Optional parameters  
(See also Programming Language Enhancements, Optional Parameters.)

The following additional enhancements are provided:

- The number of possible field levels in a data area has been increased from 9 to 99.
- It is now possible to define object handles also within a global data area. With Natural Version 3.1, object handles could only be defined within a local or parameter data area.
- It is now possible to define initial values for object handles.

The user interface has been adapted to support the features described above.

The .E line command (used to define initial values and edit masks) has been enhanced. On the separate screen that is displayed after the .E line command has been entered, the new command code A has been added to allow for the definition of long indices.

New features not available with Natural Version 3.1 require that the data area is stored in the FUSER system file using a new and extended source format. As long as no new features are used, data areas are stored using the old Natural Version 3.1 compatible source format by default to allow for sharing the data area between a Natural Version 3.1 and Natural Version 4.1 environment.

The V31COMP compiler option may be used to ascertain that a data area that is edited and cataloged with the Natural Version 4.1 data area editor can still be cataloged with Natural Version 3.1.

## Map Editor

The following Natural Version 4.1 features have been added to the Natural map editor:

- **Support of Long and Dynamic Variables**

Long and dynamic alpha variables can be used in the map editor. This includes the feature that the length of these variables will be automatically adjusted by the map editor to the length fitting on the map by the use of the AL parameter.

- **Support of Large Arrays**

The array dimensions have been adjusted to the new limits available in the Natural 4.1 system.

- **Support of Selection Boxes**

The new feature "Selection Boxes on Mainframe" has been added to the map editor. In the Extended Field Editing Menu, it is only required to specify the name of the selection-box array, the rest is done automatically by the map editor.

## NATLOAD/NATUNLD Utility

The NATLOAD/NATUNLD utility provides the following enhancement:

- If objects are to be unloaded/loaded via Entire Connection and Entire Connection has not been activated, the terminal command %+ is automatically issued to activate Entire Connection.
- A new user exit LOADEX03 (source code L-S-EX03) has been provided which can be invoked as soon as NATLOAD processing has finished.
- A new option FIXEDLENGTH in the unload-with-clause causes all records to be written to the work file with a fixed length of 252 bytes.
- A new option VERSIONCHECK in the load-object-with-clause can be used to make sure that the version of the cataloged objects to be loaded is compared with the current Natural version.
- New online commands for report processing have been introduced (see Condition Codes and User Exits in Batch Mode in the NATLOAD/NATUNLD documentation).

## SYSBPM Utility

The SYSBPM utility provides the following changes and enhancements:

- The utility main menu and the LIST function were subject to major changes.
- New utility direct commands WRITE, SORT, DISPLAY CDIRECTORY have been added.
- The direct command SELECT BUFFERPOOL is now available as a function that can be selected from the SYSBPM menu. This also eliminates the need for using the DISPLAY BUFFERPOOL and RESET BUFFERPOOL direct commands (the display function is implicate and a reset can be accomplished by selecting a different buffer pool from the selection list).
- The following utility direct commands have been changed:
  - DISPLAY CLIST replaces DISPLAY CINDIVIDUAL.
  - DISPLAY HEX replaces DISPLAY OBJECT.
  - DISPLAY LIST replaces DISPLAY INDIVIDUAL.
- A new SYSBPM user interface provides the option to sort the objects in the buffer pool by various criteria (for example, use count) in batch mode.

## SYSDDM Utility

The SYSDDM utility provides the following enhancement:

- Support of large und dynamic variables for DB2 DDMs.
- Support of access to DB2 tables on UNIX platforms.  
For details, see SQL Services in DDM Generation in the Natural for DB2 documentation.

## SYSMAIN Utility

The SYSMAIN utility provides the following enhancement:

- While the function Find Programming Object is being executed, SYSMAIN displays a window indicating the name of the object being searched for. With Natural Version 4.1, this window will also indicate whether the source form or cataloged form of an object is being searched for.
- A new online command FINDFIRST has been introduced that causes the FIND function to stop when the first object is found. See Commands Issued to SYSMAIN in the SYSMAIN documentation for more information.
- In user exit MAINEX11 (source SM-UX-11), a new flag has been introduced that enables suppression of the FIND Objects window in the FIND function.

## SYSNCP Utility

The user exits of the SYSNCP utility have been modified. Please use these new user exits and adapt them to your requirements. See also Utility Activation.

## SYSARM Utility

The SYSARM utility provides the following enhancement:

- You can execute SYSARM utility functions in direct-command mode or batch mode: see Direct Commands and Batch Processing in the Natural SYSARM Utility documentation.
- With Natural 3.1, you could only maintain parameter profiles stored on and applying to the current FNAT system file. With Natural 4.1, it is also possible to maintain parameter profiles of other FNAT system files.
- You can enter a commentary text in each editor line, or spread a comment over as many lines as you like.

## SYSRDC Utility

The SYSRDC utility provides the following enhancements:

- New function Trace Record by Number (function code N)
- New function to show the status of trace recording (function code C)

## SYSTP Utility

The SYSTP utility provides the following enhancement:

- A new SYSTP function allows you to cancel Natural user sessions depending on their last-activity date.

## SYSWEB Utility

A new version of SYSWEB based on Natural for Windows/UNIX Version 6.1 has been provided.

## Utility Activation

From Natural Version 4.1, Natural invokes a Natural utility without performing a logon to the corresponding utility library in the FNAT system file. This applies to all utilities listed in Support of Activation without Logon below. As a result, Natural preserves the global data area (GDA) and/or application-independent variables (AIV). For resetting the GDA and/or the AIVs, see the parameter FREEGDA below. The current user library and the steplib settings are maintained.

To preserve the settings of your application environment, do **not** log on to a utility library. Instead, invoke a utility by using the Natural system command that corresponds to the utility.

After terminating a utility, you will be returned to the library from which you invoked the utility. However, if you explicitly log on to a utility library before invoking the utility, you will stay in this (utility) library after utility termination.

The behavior described above can have an impact on Natural Version 4.1 batch procedures that contain logon instructions to utility libraries.

For information on how to control access to a utility in a Natural Security environment, see Protecting Utilities in the Natural Security documentation.

Below is information on:

- Support of Activation without Logon
- Source Editing
- FREEGDA Parameter
- SYSNCP User Exit

## Support of Activation without Logon

Below are the Natural utilities that support utility activation (without logon to the corresponding utility library) by using the corresponding Natural system commands:

- NATLOAD
- NATUNLD
- SYSADA
- SYSBPM
- SYSDB2
- SYSDDM
- SYSEDT
- SYSERR
- SYSMAIN
- SYSNCP
- SYSAFOS
- SYSSAFOS
- SYSOBJH
- SYSPARM
- SYSPPOOL
- SYSRPC
- SYSSEC
- SYSSQL
- SYSTP (including SYSMON, SYSSWAP, SYSFILE, BUS)
- SYSTRANS
- SYSUNLD



Below are the Natural utilities/Natural system commands that do **not** support utility activation without logon to the utility library. Invoking these utilities will still cause an implicit logon to the utility library and user settings can be lost.

- SYSEXT
- SYSEXV
- SYSRNM (to apply the utility activation without utility library logon requires a new Review Natural Monitor version; see Natural and Other Software AG Products in the section General Information of these Release Notes)
- All other Natural add-on and Software AG products using the Natural FNAT system file.

## Source Editing

Source editing in utility libraries still requires a logon to the corresponding utility library.

## FREEGDA Parameter

The profile parameter FREEGDA controls whether current user global data area (GDA) and application-independent variables (AIV) are to be reset when a utility is invoked by using the Natural system command that corresponds to the utility's name.

Note that the preservation of the current user GDA and AIV variables will increase the data size accordingly and can lead to thread size problems under certain operating systems. In this case, you can use the system command LOGON *library* to force a logon to the utility library.

See also FREEGDA in the Natural Parameter reference documentation.

## SYSNCP User Exit

With utility activation under Natural Version 4.1, User Exit NCP-USR1 has changed. If you apply this user exit, use the new NCP-USR1 program and adapt it to your requirements.

# Application Programming Interfaces USR\* in Library SYSEXT

With Natural Version 4.1, the USR\* programs (APIs, previously named "user exits", see also note on terminology change in the section Documentation) from the library SYSEXT delivered will run in a special mode. As a result, the USR\* programs need no longer set further steplib to execute related Natural objects for processing. This will reduce the impact on the Natural buffer pool search logic and will improve performance significantly if application programming interfaces are used extensively within user written applications.

To introduce this mode, the application programming interfaces must be cataloged with Natural Version 4.1. This implies that the application programming interfaces cannot be executed with Natural Version 3.1. This behavior is different to the previous Natural version and will have some impact on the migration path of applications that call application programming interfaces.

## Use of USR\* Programs

Usually the access of USR\* programs by an application requires that the application programming interfaces are copied from the library SYSEXT to the application libraries on the FUSER system file or to the library SYSTEM on the FUSER system file or the library SYSTEM on the FNAT system file or any other library which is defined as steplib for the application. Library SYSEXT can also be used as steplib. Because, as of Natural Version 4.1, the application programming interfaces delivered will always be cataloged with the latest Natural version, we recommend that the application programming interfaces should reside on the FNAT system file. This will ensure that the right version is executed and separates user-written applications from modules provided by Software AG.

If applications that call application programming interfaces should run with both Natural Version 3.1 and Natural Version 4.1, the application programming interfaces delivered with the corresponding Natural version must be used.

The following scenarios may be considered:

- Using a Version 3.1 FUSER File for Natural Version 3.1 and 4.1
- Using a Version 3.1 FUSER File for Natural Version 4.1 only
- Using a New FUSER File for Natural Version 4.1
- Migration

## Using a Version 3.1 FUSER File for Natural Version 3.1 and 4.1

If the same FUSER system file is to be used in a Natural Version 3.1 and Version 4.1 environment in parallel, the following steps are recommended:

- Remove all USR\* modules you have copied from the library SYSEXT into application libraries on your FUSER system file.
- In both environments, copy the used USR\* modules from the library SYSEXT to the library SYSTEM on the corresponding FNAT system file.

Alternatively, the USR\* modules can be moved to another system library on FNAT, which then must be defined as steplib, or the library SYSEXT can be used as steplib for the applications. Then, automatically in both environments, the right versions of the application programming interfaces are executed.

## Using a Version 3.1 FUSER File for Natural Version 4.1 only

If you want to use the existing Natural Version 3.1 FUSER system file and you do not want to share the FUSER system file, then it is still possible to replace all USR\* modules you have copied from the library SYSEXT into application libraries with the new USR\* objects from the Version 4.1 library SYSEXT.

However, the preferred way is to remove all application programming interfaces on the FUSER system file and copy the used application programming interfaces from the library SYSEXT to the library SYSTEM of the FNAT system file or use an SYS library on FNAT as steplib.

## Using a New FUSER File for Natural Version 4.1

If you want to port existing applications to a new FUSER file, copy all applications objects but no USR\* objects originating from Software AG to the new FUSER system file. Then proceed as described in the scenario above.

## Migration

The function FIND of the SYSMAN utility can be used to search for all USR\* modules stored in a specific library on the FUSER system file or across the whole system file. In addition, Predict cross-reference data can be used to determine all referenced application programming interfaces.

## New Application Programming Interfaces in Library SYSEXT

The library SYSEXT provides the following new application programming interfaces (APIs, previously named "user exits", see also note on terminology change in the section Documentation):

API	Description
USR2035	Set SSL parameter string for communication via RPC.
USR2073	Ping or terminate an RPC server.
USR4001	Set Natural profile parameter PROGRAM dynamically.
USR4002	Retrieve variables of the current system.
USR4003	Retrieve Natural stack information.
USR4004	Retrieve Natural dynamic parameter.
USR4005	Read all current key settings.
USR4007	Get/set current value of profile parameter SYNERR.
USR4340	Natural Buffer Pool Interface 4340.
USR6002	Get the current values of some internal counters.

## Software AG Editor

- The Software AG Editor parameter macro NTEDBP is now an integral component of the parameter module NATPARM.
- The Software AG language support has been changed. The language tables (English, Hebrew and Japanese) were moved from the macro NATEDTAB into separate macros NTEDTAB $n$  ( $n=1,2,3$ ) which are called by the source module NATCONFIG.
- The Software AG Editor supports the Natural system variable \*LANGUAGE for Hebrew (ULANG=33) and Japanese (ULANG=59).
- The Software AG Editor supports the changes of format for NODE and JOB-NUMBER introduced with Entire System Server Version 3.2.1.

## Debugger Enhancements

- Enhanced Display Functions
- Break Processing
- Commands

### Enhanced Display Functions

- The List Source function in addition supports the display on Model-4 screens.
- In the List Source function, you may set a watchpoint by setting the cursor on a variable and pressing PF18.
- The Display Variables screen was enhanced with a zoom function (PF5) to display large variable names, possibly with long indices.
- The screen for displaying the content of a single variable was enhanced to display long variables in chunks of 256 bytes after a selected position. Browsing in the content of a long variable is possible with PF keys PF22/PF23.

### Break Processing

- Watchpoints can be set for long or dynamic variables. If the length exceeds 253 bytes, the comparison value of a watchpoint is restricted to the first 253 bytes.
- New watchpoint type: persistent watchpoint.  
A persistent watchpoint allows extending the scope of investigation of a variable value: whereas a standard watchpoint checks for a change of content of a variable in the scope of an object, a persistent watchpoint also checks the change of content of a variable on a subordinate level.
- New watchpoint operator: INV.  
The watchpoint operator INV allows you to check any invalid content of variables of the type N, P, D, T.

### Commands

- The command DBLOG is now also available during debugging.

# Database Interfaces

The following topics are covered:

- Natural for Adabas
  - Natural for DB2
  - Natural for DL/I
  - Natural for VSAM
-

## Natural for Adabas

The following enhancements are provided with Natural for Adabas:

- Support of Adastar Discontinued
- Adabas-Related Natural Enhancements

### Support of Adastar Discontinued

Natural Version 4.1 requires at least Adabas Version 7.1. Consequently, Adastar will no longer be supported.

### Adabas-Related Natural Enhancements

The following Adabas-related enhancements are provided with Natural Version 4.1:

- Dynamic change of reading direction within an active READ or HISTOGRAM processing loop.
- Dynamic repositioning within a READ processing loop.
- New comparators LESS/GREATER THAN and LESS/GREATER EQUAL for READ and HISTOGRAM statements.
- Support of Multi-Fetch in FIND, HISTOGRAM and READ statements.
- New keyword TO that enables end of range condition (ENDING AT) to be controlled by the database itself.
- Support of the Adabas Transaction Manager.

## Natural for DB2

The following enhancements are provided with Natural for DB2 Version 4.1:

- The SQL syntax enhancements provided by DB2 Versions 6 and 7 will be supported.
- With earlier versions of Natural for DB2, you have to use Natural Format A for SQL data types DATE and TIME. With Version 4.1, it is possible to use Natural variables of Formats D and T; these are automatically converted to SQL data types DATE and TIME.
- Large objects (alphanumeric and binary variables of up to 1 GB) are supported.
- In addition to the two existing DB2 parameter styles, the new parameter style "DB2SQL" for the passing of data to stored procedures are supported.
- It is possible to write "user-defined functions" in Natural.
- Support of insensitive and sensitive static scrollable cursors in the Natural SQL statement SELECT.
- New comparators LESS/GREATER THAN and LESS/GREATER EQUAL for READ and HISTOGRAM statements.
- New keyword TO that enables end of range condition (ENDING AT) to be controlled by the database itself.
- Dynamic change of reading direction within an active READ or HISTOGRAM processing loop.
- Allow usage of SQL reserved words as SQL identifiers by enabling delimited identifiers generation. (See NDBPRM parameters DELIMID and RWRDONL.)
- The SYSDDM utility supports the definition of DDMs to enable access to DB2 tables on UNIX platforms. For details, see SQL Services in DDM Generation in the Natural for DB2 documentation.

The Natural Tools for DB2 provide the following enhancements:

- SYSDDM SQL Services
  - Support of new DB2 data types by DDM fields with a length of up to 1 GB -1 byte or with dynamic fields.
- ISQL
  - Infrastructure for "SQL skeleton members" for SQL statements, which are not supported by the catalog maintenance.

- Model 4 support.
- Environment Setting
  - Support of four additional special registers.
- Catalog Maintenance
  - Support of the new DB2 data types in the CREATE TABLE statement.

## Natural for DL/I

- DL/I-Related Enhancements
- Note Concerning NDB / NSB Generation in LE370 Environment

### DL/I-Related Enhancements

The following DL/I-related enhancements are provided with Natural for DL/I Version 4.1:

- Dynamic repositioning within a READ processing loop.
- New comparators LESS/GREATER THAN and LESS/GREATER EQUAL for READ and HISTOGRAM statements.
- New keyword TO that enables end of range condition (ENDING AT) to be controlled by the database itself.

### Note Concerning NDB or NSB Generation in LE370 Environment

If you are not using SMA:

When executing modules NDPBNDB0 or NDPBNSB0 in an LE370 enabled batch Natural to store an NDB or NSB into the FDIC system file, the following Natural error message may be issued:

```
SYSDLI 3970 Error when loading NDB/NSB
```

The step ends with Condition Code 8 in this case.

To prevent this error, the NDB or NSB load module must be link-edited with AMODE(31).

The binder step will then end with Return Code 4 due to the following warning message:

```
IEW2651W 511C ESD AMODE 24 CONFLICTS WITH USER-SPECIFIED AMODE 31
```

This return code must be ignored in the following step by means of a COND=(8,LE) keyword.



## Natural for VSAM

The following VSAM-related enhancements are provided with Natural for VSAM Version 4.1:

- Dynamic change of reading direction within an active READ or HISTOGRAM processing loop.
- Dynamic repositioning within a READ processing loop.
- New comparators LESS/GREATER THAN and LESS/GREATER EQUAL for READ and HISTOGRAM statements.
- New keyword TO that enables end of range condition (ENDING AT) to be controlled by the database itself.
- Support of native Smarts environment for Natural Development Server.

A new parameter SMARTS for the IO Interface NVSMISC is provided to support Complete and/or Smarts environments.

The function MIGRATE for the parameter SFILE in the NVSPARM module has been dropped.

# Operating System and Teleprocessing Interfaces

The following changes/enhancements to operating-system interfaces and teleprocessing-system interfaces are provided with Natural 4.1:

- Natural for VSE/ESA
  - Natural for BS2000/OSD
  - Natural Com-plete/SMARTS Interface
  - Natural IMS/TM Interface
  - Natural CICS Interface
  - Natural CICS Interface and Natural UTM Interface
  - Support of New 64-Bit Architecture
  - Improved Abend and Error Handling
  - Natural Roll Server
  - Authorized Services Manager
  - SMARTS Print/Work File Support
-

## Natural for VSE/ESA

Natural Version 4.1 offers VSE library support for all input, print and work files for batch Natural under VSE/ESA.

## Natural for BS2000/OSD

- New Keyword Parameters
- Changed Keyword Parameter

### New Keyword Parameters

The following keyword parameters have been added to the macro ADDON and the programs CMPSTART and PREFRESH:

Keyword Parameter	Purpose	Used in:
LOAD=BIND/ASHARE	Defines the method for loading a module into a common memory pool.	ADDON, CMPSTART and PREFRESH
JV=YES/NO	Controls via a Job Variable whether a global common memory pool is successfully enabled.	CMPSTART
JVSUFX=xxxxxxxx	Suffix of the Job Variable name.	CMPSTART

### Changed Keyword Parameter

The keyword parameter REQMLOC now applies to the generation of both the front-end and reentrant parts of the BS2000/OSD batch and TIAM drivers.

## Natural Com-plete/SMARTS Interface

- New Functionality
- Generation Parameter Changes
- Change in Installation

### New Functionality

- Natural Version 4.1 supports Com-plete Version 6 SMARTS/HTTP server environments.
- Full support of native SMARTS environment (APS Runtime).
- The local editor buffer pool server routine has been enhanced. It features new error message handling, changed initialisation und termination handling as well as a command interface.
- The new DLL NCFBTIO enables Natural to use the SMARTS Portable File System (PFS) as well as native datasets as a container for batch IO. It is activated with parameter TTYPE=BTCH or with the terminal command %T=BTCH when Natural is running in a SMARTS Server environment.
- The Natural Com-plete/SMARTS Interface is a Software AG product with a new product code NCF.

In the Natural Com-plete/SMARTS Interface, the following Natural Com-plete error messages have been replaced by Natural error messages:

Natural Com-plete Error Messages	Replaced by Natural Error Message
NCF0001	NAT9915
NCF0002	NAT9907
NCF0003	NAT9996
NCF0004	NAT9981
NCF0005	NAT9986
NCF0006	NAT9980
NCF0007	NAT9991
NCF0008	NAT9909
NCF0009	NAT9996

### Generation Parameter Changes

The old macro NCMCFPRM has been replaced with a new macro NFMPRM.

The following macro parameters have been dropped:

Macro Parameter	Purpose
ADDBUF	Size of additional IO buffer
CRELO	Defines the table of resident page programs which are relocatable.
EDITWRK	Specifies parameters for editor work files to be able to use the Software AG Editor.
NUCRELC	Specifies whether a nucleus relocation check takes place or not.

Instead of the dropped parameter EDITWRK, you can use the Natural profile parameter EDBP to set the relevant parameters. The new Natural Server in Com-plete NCFNAT41 uses the buffer pool sizes from the Natural profile parameter BPI.

The following macro parameter is new:

Macro Parameter	Purpose
SERVER	Defines the name of the Natural server which is initialized during Com-plete startup.

## Change in Installation

In the Com-plete installation, the link process for the front-end part and for the Natural Server has been changed. The Natural Com-plete modules, sources and macros are delivered in new separate libraries instead of in Natural standard libraries.

## Natural IMS/TM Interface

- New Functionality
- Error Codes Replaced by Natural Error Messages

### New Functionality

- Support of user written and language-specific LE condition handlers in case of abends.
- Support of customer-specific subpool number for all working storage requests.
- Reintroduction of the TRNCODE parameter that allows you to define a logical transaction code which is used by the BMP, NTRD and SRVD drivers.
- Optional elimination of thread relocation to support Con-form.
- Support of logical NSB names defined in the transaction code table NIITRTAB.
- Support of Natural Access Method AM=STD by the NTRD and SRVD drivers
- Support of special values \*WTO, \*MTO and \*PRINT $nn$  for the SENDER destination used by the NTRD and SRVD drivers.
- Default alternate TP PCB number in the NIMTRNTG macro that is used by the service modules CMQUEUE, CMQUEUEEX and NIIDQUMS.
- Support of a customer-specific alternate TP PCB number in the service modules CMQUEUE, CMQUEUEEX and NIIDQUMS.
- Support of multiple Natural/IMS servers in the same MPP.

### Error Codes Replaced by Natural Error Messages

In the Natural IMS/TM Interface, the following Natural IMS/TM interface error codes have been replaced by Natural error messages:

Natural IMS/TM Error Code	Replaced by Natural Error Message
3504	NAT9996
3514	NAT9902
3615	NAT9903
3616	NAT9904
3617	NAT9986
3628	NAT9974

In addition, the following changes were made:

<b>Natural IMS/TM Error Code</b>	<b>Replaced by</b>
3622	an NUS message corresponding to the return code.
3660	an NUS message corresponding to the return code.

The error codes in the section "Natural IMS/TM Error Codes" in the Natural Messages and Codes documentation have been removed accordingly.

## Natural CICS Interface

- New Functionality
- Generation Parameter Changes
- Change in Installation

### New Functionality

- In addition to work files under CICS, the Natural CICS Access Method now also supports print files under CICS.
- Natural as a Server is now also supported in a CICS environment.
- Natural parameter logging is supported in Natural under CICS, that is, the PLOG parameter is honored now.

### Generation Parameter Changes

NCMDIR macro in NCISCPCB:

The following macro parameters have been dropped:

Macro Parameter	Purpose/Reason
CDATE	These parameters are obsolete, because only High Level Assembler is supported now.
CTIME	
ROLLFLS	Maximum Number of VSAM Roll Files Now a maximum of 9 VSAM roll files is always taken.
SWPSIZE	Swap Pool Size The specification of the Natural swap pool under CICS is now done using the NTBPI generation macro within NATPARM or the profile parameter BPI; in addition to the swap pool size, also a swap pool cache size and a swap pool name may be specified in that way.

The following macro parameter has been changed:

Macro Parameter	Purpose/Reason
CICSPLX	Switching of CICS Application Region This parameter is now available with new parameter values.

The following macro parameter is new:

Macro Parameter	Purpose/Reason
SIPSERV	Authorized Services Manager's Session Information Pool This parameter is now available for SIP server usage.



NCMPRM macro in NCIPARM:

The following macro parameters have been dropped:

Macro Parameter	Purpose/Reason
ASA	Use ASA Control Characters For Natural Message Logging  Now NCI determines internally whether ASA control characters for the NCI message log destination have to be provided or not.
CDATE	These parameters are obsolete, because only High Level Assembler is supported now.
CTIME	

The following macro parameter in the macro NCMPRM has been changed:

Macro Parameter	Purpose/Reason
FLDLEN	Supply Field Length List On External Program Call  Two sub-parameters are available now.

The following macro parameters are new:

Macro Parameter	Purpose/Reason
DUPTID	Handle Duplicate Terminal ID
MSGPFX	Generate NCI Message Prefix for WTL Messages
UNITID	Establish Unique Terminal IDs

## Change in Installation

The Natural CICS interface programs hold all CICS commands in unexpanded form, that is, all CICS commands require a pre-compile step. If you are not using the most recent CICS version, the pre-compile step may result in a non-zero return code because of CICS commands used that are unknown to your preprocessor; however, the subsequent assemble step skips the flagged CICS commands using conditional assembly.

## Natural CICS Interface and Natural UTM Interface

### New Keyword Subparameters for Macro NTSWPRM

It is possible to define ESA Data Space in addition to a Swap Pool (which, under BS2000/OSD, is a "global" swap pool). For swap pool definition, new keyword subparameters have been added to the macro NTSWPRM.

Keyword Parameter	Purpose
DSPCONT= <i>nn</i>	Minutes for Data Space slot controlling.
DSPLIFE= <i>nn</i>	Lifetime (minutes) of a thread in the ESA Data Space.
SWPFILE=FNAT/FUSER	System file definition for the swap pool initialization parameters.

### Support of New 64-Bit Architecture

Under the z/OS operating system, Natural 4.1 supports IBM's new 64-bit architecture. This applies to the Natural roll server (for cache), the Natural Authorized Services Manager (ASM) (for cache), and the buffer pool (e.g., for server Natural under OS/390), all of which at present use data spaces as local cache. With Version 4.1, you are able to choose between data spaces and memory objects to be used as local cache for these parts of Natural.

### Improved Abend and Error Handling

A new profile parameter DUE and a new terminal command %DUE have been provided to enable the generation of a snap dump whenever a specific Natural error occurs. This will facilitate error diagnosis, as it will not require the application of so-called "trap Zaps".

A new setting DU=ABEND has been provided to produce a memory dump with the original abend reason code at session termination. For DU=ON a non-zero reason code is displayed by termination message NAT9974 after the abend code.

In a case of an abend, the part of the Natural Buffer Pool holding the Natural object that is currently executed and the corresponding directory is always included in the dump.

LE/370 is supported also for VSE/ESA and CICS.

The LE abend handling has been improved. New error messages NAT0950 and NAT9967 inform about conditions that could not be handled by Natural. Additional diagnose information is printed depending on the LE runtime option TERMTHDACT.

## Natural Roll Server

The Roll Server is now contained in a single module, NATRSM41. This module incorporates all necessary subtasks that were shipped in separate modules with Version 3.1. The former user exits NATRSUX1 and NATRSUX2 have been renamed to NATRSU14 and NATRSU24, respectively. The parameter list for NATRSU14 has been expanded to include information about total number of slots and number of slots in use for each roll file. Please refer to Roll Server User Exits in the section Natural Roll Server Operation of the Natural Operations for Mainframes documentation.

The roll file formatting routine, NATRSRFL, now has a new function, FORMAT, that has a simpler parameter structure than the former CINIT function, see Natural Roll Server Operation of the Natural Operations for Mainframes documentation.

## Authorized Services Manager

The Authorized Services Manager is now contained in a single module named NATASM41. This module must be linked with the module NATBPMGR at the customer site, see Installation Procedure under OS/390.

A special module, NATAU31B is provided for IMS customers who wish to use the Natural IMS/TM Interface Version 4.1 together with the Natural IMS/TM Interface Version 2.3 in the same IMS/TM MPP environment, see Migration Hints in the section General Information of these Release Notes.

## SMARTS Print/Work File Support

The Software AG Multi-Architecture Runtime System (SMARTS) offers an operating system interface common for all supported platforms. Beside the necessary features for server environments (Multi-Tasking, resource locking), it also supports access to a portable file system and printer support.

This offers the possibility to enable print and work file access for Natural servers running under SMARTS. This is of particular importance for the Natural Development Server (NDV) to allow an NDV client to test programs even if they contain print/work file access.

# Natural XML Toolkit

With Natural Version 4.1.2 for Mainframes, the XML Toolkit has been enhanced by the functionality provided in Natural for Windows/UNIX Version 6.1.

The following topics are covered:

- Purpose of the Natural XML Toolkit
  - XML Toolkit on the Mainframe
  - Natural XML Toolkit Functions
  - Mapping Natural Data Definitions to DTDs
  - Serializing the Data to XML
  - Mapping DTDs to Natural Data Definitions
  - Parsing the XML File and Assigning the Contents to a Natural Data Structure
-

## Purpose of the Natural XML Toolkit

The Natural XML Toolkit is a set of tools for XML processing that provides functionality required for the integration of XML processing into Natural. It improves the integration of Natural applications with XML, without requiring external software products.

The XML Toolkit can be considered an intermediate step towards the integration of XML processing into Natural. The next step would be the full integration of XML functionality into the Natural programming language.

The XML Toolkit consists of a set of Natural programs, some of which are available in source-code form. The XML Toolkit programs may be integrated into your Natural applications, thus providing access to XML data or supplying data from Natural in XML format.

## XML Toolkit on the Mainframe

Natural for Mainframes does not support the REQUEST DOCUMENT statement. Therefore, XML documents have to be processed using Natural work file processing.

If, however, a computer running Natural Version 5 on UNIX, OpenVMS or Windows is accessible on the network, the REQUEST DOCUMENT statement can be executed on this remote machine, using a Natural Remote Procedure Call.

## Natural XML Toolkit Functions

The XML Toolkit provides the following functions:

- **Mapping of Natural data definitions to DTDs, and vice versa.**  
DTDs (document type descriptions) are most commonly used to describe the structure of an XML document.
- **XML Token => Natural Data**  
After creating the Natural data structure, the XML document has to be parsed and saved into the data structure. An implementation for the XML "tokenizer" delivered will be generated. This tokenizer assigns the value of a data element to the corresponding data structure.
- **Natural Data => XML Document ("Serialization")**  
Serialization is the process of taking the data stored in the Natural data structures and creating an XML document according the description in the DTD.
- **Check data structure for:**
  - alternative elements,
  - range of values for attributes,
  - occurrences and boundaries of elements.

## Mapping Natural Data Definitions to DTDs

The mapping of Natural data definitions to a DTD is the first step to bind Natural data structures to XML tags, and is required to implement a representation of Natural data as XML tags. The generated DTD is used later during the serialization to an XML document.

## Serializing the Data to XML

During the execution of a Natural program, the fields defined in the DEFINE DATA statement are filled with real contents. During serialization, these contents are written to a dynamic variable in XML format using the previously generated DTD as input. The XML Toolkit also generates the program for the serialization of the data.

## Mapping DTDs to Natural Data Definitions

The mapping of a DTD to Natural data structures shows the differences between the two. As the DTD does not specify how many records are to be included in the XML document, the XML Toolkit assumes a maximum number to be included. The application programmer may know the precise number and can adapt the data structure accordingly. A similar limitation exists with the length of the data: the DTD does not contain information about the length of the data in a record; therefore, the XML Toolkit generates fields in the data structure with a length of 253 (the current maximum length).

## Parsing the XML File and Assigning the Contents to a Natural Data Structure

The XML Toolkit generates Natural code based on the DTD. This code represents a subroutine which is invoked from the Toolkit's "tokenizer" to assign the contents of the tags in the XML document to the corresponding Natural data structure.

# Miscellaneous Changes and Enhancements

The following topics are covered:

- Profile Parameters
  - NATCONFIG Enhancements
  - Natural Performance Enhancements
  - Source Change Inventory
  - Delivery of Source Changes Enhanced
  - Non-Sliding Year Window (YSLW Parameter)
  - Stack
  - Buffer and Storage Management
  - Natural Global Buffer Pool
  - NATSREX2/3 Example
  - Reduction of Object Code
  - Change/Enhancements Requests
  - Natural Web Interface
-

## Profile Parameters

- New Profile Parameters
- New Parameter Macro
- Enhanced Profile Parameters
- Profile Parameters Dropped
- Retrieving Dynamic Profile Parameters
- Parameter Macro NTFIELD
- Profile Parameter PLUGIN

### New Profile Parameters

The following new Natural profile parameters are provided with Natural Version 4.1:

Profile Parameter	Description
BPC64	Specifies the type of storage for the local Natural buffer pool. It corresponds to the C64 subparameter of the BPI profile parameter or the NTBPI macro.
BPCSIZE	Defines the Size of the local Buffer pool cache. It corresponds to the CSIZE subparameter of the BPI profile parameter or the NTBPI macro.
BPMETH	Specifies the search algorithm that is to be used for allocating storage in the Natural buffer pool. It corresponds to the METHOD subparameter of the BPI profile parameter or the NTBPI macro.
DS	Defines the default initial size of various Natural storage buffers (for example, ASIZE, DATSIZE). A corresponding new macro NTDS is also available.
DUE	Allows improvedabend and error handling.
EDBP	Controls the initialization and operation of the Software AG Editor buffer pool and its work file.
EMFM	With this parameter, you can activate/deactivate the Edit Mask Free mode at session startup. This mode allows you to omit literals during input into a field with a numeric edit mask.  Within a running Natural session, you may override this setting with the new terminal command %FM+ or %FM-.
FREEGDA	Determines whether or not the currently used global data area is to be kept when a Natural utility is invoked, that is, whether the global data area is to be still available after the use of the utility or to be newly initialized.
OVSZIE	Defines the maximum total amount of variable storage which can be allocated outside the Natural thread. See Buffer and Storage Management below.
STACKD	Defines the character to be used as the command delimiter for the STACK parameter and for command input under the Natural Development Server (NDV) in a Natural Single Point of Development environment.
TMODEL	Defines the type of terminal model (screen size) being used. This is relevant for environments like IMS/TM, where the TP monitor is not capable of automatically supplying Natural with this information.



## New Parameter Macro

The following new Natural parameter macro is provided with Natural Version 4.1:

Parameter Macro	Description
NTCSTAT	This parameter macro has been introduced to break the limit of 256 bytes for the CSTATIC profile parameter. With the new macro you can define external subprograms to be statically linked to the parameter module.

## Enhanced Profile Parameters

The following Natural profile parameters have been enhanced or changed for Natural Version 4.1:

Profile Parameter	Enhancement
ADAMODE	With Natural Version 3.1, if one of the values 1, 2 or 3 is specified for the ADAMODE parameter, but an Adabas subcomponent is unable to perform an Adabas X48 communication, ADAMODE is set to 0 and the session is continued without an error message. With Natural Version 4.1, an error message is issued before ADAMODE is set to 0. The session is continued if ITERM=OFF has been specified, otherwise the session will be terminated.
BPI	New subparameters: <ul style="list-style-type: none"> <li>● CSIZE=nnnn to set the size of the cache for the local buffer pool.</li> <li>● METHOD= to determine the algorithm for allocating storage in the buffer pool.</li> <li>● C64= to determine the type of storage to be used for the buffer pool cache, either 64-bit memory objects or a data space.</li> <li>● TYPE=SWAP to define a Buffer pool to hold the Natural CICS swap pool.</li> </ul>
CMPO	Same as for COMPOPT system command.
CSIZE	The range of possible values for the Connect buffer area size has been changed from 0 - 128 to 0 - 512 KB.
DATSIZE	As of Version 4.1.2, this parameter is used only to set the minimum size of the local data buffer (DATSIZE).  <b>Note:</b> With previous versions of Natural, this parameter could be used to set the minimum <b>and/or the maximum</b> size of the DATSIZE buffer: DATSIZE=(min,max). The maximum setting was dropped for design considerations. The profile parameter OVSIZE can be used instead to limit the memory allocation outside of the thread. A limitation of individual buffer sizes is not provided.
DU	New setting DU=ABEND to produce a memory dump in the case of an abnormal Natural session end and terminate the session with an abend code (same as DU=ON, except that with DU=ON the session is terminated with an error message). This is particularly relevant under IMS/TM.
LT	With Natural Version 3.1, LT=0 disallowed any database access. This made the setting LT=0 useless. With Natural Version 4.1, LT=0 indicates that no limit is to be in effect, so that LT is consistent with other limits, such as MADIO and MAXCL.

PRINT	<p>New subparameter PADCHRO='y' to define the padding character to be used for output datasets on the print file. The default value is PADCHRO=' ' (blank).</p> <p>New subparameter ROUTE=xxx to determine whether logical print file routing is done according to the OUTPUT clause of the DEFINE PRINTER statement.</p> <p>New subparameter ASA=xxx for all AM=STD routines to determine whether the ASA record format is used.</p> <p>You can now define the hardcopy printer as Printer 0 for all environments.</p>
RPC	The profile parameter RPC and the parameter macro NTRPC have been changed/enhanced, see Changes/Enhancements to Profile Parameter RPC in the Natural RPC section of these Release Notes.
SORT	You can define up to two additional options for the external sort program in EXTOPT. The options are delimited by a slash (/).
TD	<p>New settings:</p> <ul style="list-style-type: none"> <li>● TD=(+/-hh,mm,ss) - In addition to hours and minutes, it is possible to specify seconds for the time differential.</li> <li>● TD=timezone - Name of the time zone to be used. This must be defined as a valid time zone in the NTTZ macro of the NATCONFIG module.</li> </ul>
TS	<p>With Natural Version 3.1, the translation of system library output is performed for the entire contents of the page buffer immediately before being sent to the screen.</p> <p>With Natural Version 4.1, the translation will be performed individually for each field at the time when it is written into the page buffer.</p> <p>For more information, see Translation of System Library Output.</p>
WORK	<p>New subparameter settings:</p> <ul style="list-style-type: none"> <li>● PADCHRI='x' to define the padding character to be used for input datasets on the work file. The default value is: PADCHRI=' ' (blank).</li> <li>● PADCHRO='y' to define the padding character to be used for output datasets on the work file. The default value is: PADCHRO=X'00'</li> </ul>
WPSIZE	New settings for a <i>maximum_below</i> and a <i>maximum_above</i> size to limit the total physical storage in KB which can be allocated below and/or above the 16 MB line.
YSLW	New settings to specify a certain year for the definition of a non-sliding year window.

## Profile Parameters Dropped

The following Natural profile parameters (and parameter macros) have been dropped as of Natural Version 4.1:

- DCOM (NTDCOM) - NaturalX Support
- NTBPI TYPE=DCOM
- IDSIZE - Size of Buffer for Natural/IDMS
- WSIZE - Size of Buffer Area for Screen Images

## Retrieving Dynamic Profile Parameters

A new application programming interface, USR4004, is available which enables you to retrieve the dynamic profile parameter settings of the current Natural session and pass them to another Natural session. This is particularly useful when a Natural session starts another asynchronous Natural session via CMTASK. The user exit will also be useful to ascertain which dynamic profile parameters have been specified for the current session.

## Parameter Macro NTFILE

The parameter macro NTFILE has been replaced by the new parameter macro NTLFILE with different syntax, but equivalent functionality.

Old: NTFILE ID=*logical filenr*,DBID=*dbid*,FNR=*filenumber*

New: NTLFILE *logical filenr,dbid,filenumber*

For compatibility reasons, the old macro NTFILE is still supported.

## Profile Parameter PLUGIN

Because all components of Natural Version 3.1 that could have been enabled using the profile parameter PLUGIN are an integral part of Natural Version 4.1, the only possible setting of the PLUGIN parameter is OFF.

## NATCONFIG Enhancements

- A new macro NTMSG can be used to define a message and a log destination in NATCONFIG for error message logging.
- A new option MIN for the NTBUFID macro can be used to define a minimum start size for a specific variable buffer.

## Natural Performance Enhancements

- The Natural Turbo Performance Plug-in of Natural Version 3.1 is an integral part of Natural Version 4.1. With this performance plug-in, loading Natural objects from the Natural System file into the buffer pool requires fewer database calls and searching for Natural objects in the buffer pool causes less CPU load.
- The compression rate of the compression algorithm has been increased; this will reduce the compressed size of a Natural thread, which, in turn, will reduce the time needed for a roll-in/out event required particularly by a terminal I/O.
- The runtime algorithms for assignments, arithmetic operations and comparisons have been enhanced.
- The Adabas Multi-Fetch option is available for database statements (see Programming Language).
- To minimize the swap pool I/Os under CICS and UTM, it is possible to generate an additional Swap Pool ESA Data Space.
- All local buffer pools and Caches can be shared in a batch and TSO environment for multiple users
- The new utility activation without an explicit LOGON preserves the current active steplib chain of the application which reduces the number of buffer pool and/or database calls if the application is continued after return from the utility.

## Source-Change Inventory

In addition to the Zap-inventory modules (e.g. NATAZAP) which exist for Natural and each of its subproducts and list all Zaps applied to the product, Natural Version 4.1 will provide source-change inventory modules in which all source changes applied to a product are logged. This enables support personnel to ascertain the precise state of your product installation, which will facilitate error diagnosis.

## Delivery of Source Changes Enhanced

With Natural Version 4.1, the delivery of single source changes as text files will be discontinued for the following product:

- Natural for CICS

Instead of text files, the affected and all prerequisite sources modules and macros will be delivered as a single package that also includes all previous fixes. This ensures that all prerequisite fixes are applied and that the respective operating system or TP monitor system interface includes all available fixes. In addition, this enhancement prevents typing mistakes that might occur when a source change is incorporated into the respective source module or macro.

## Non-Sliding Year Window (YSLW Parameter)

With Natural 3.1, the YSLW profile parameter allows you to set a so-called "year sliding window" of 100 years to relate a 2-digit year value to a specific century. The value specified with the YSLW parameter determines how many years in the past - that is, before the current year - that 100-year range is to begin. In other words, the window range is always in relation to the current year.

With Natural Version 4.1, it is also possible to set a "non-sliding" year window: You can set the year in which the 100-year range is to begin. For example, if you specify YSLW=1980, the window is from the year 1980 to the year 2079, regardless of what year the current year is.

## Stack

A new application programming interface, USR4003, is available in the library SYSEXT. It enables you to:

- ascertain the number of entries in the stack,
- ascertain the type of a stack entry (command, data),
- read a specific entry from the stack.

Thus it is possible to retrieve information from the stack without having to use an INPUT statement; that is, without the other functions of an INPUT statement being performed, as these may not be desired if merely a stack entry is to be read.

## Buffer and Storage Management

- With Natural Version 4.1, the internal I/O buffers, which are currently fixed-size buffers with a maximum size of 32 KB, are allocated as variable buffers. This avoids situations which lead to errors NAT1114 "internal output buffer overflow" and NAT1150 "attribute buffer overflow".
- Normally, Natural automatically increases the sizes of all its variable buffers (DATSIZE, etc.) as required. Natural Version 4.1 allows you to restrict this automatic increase if you wish: for each variable buffer and work pool, you can define an initialization and/or a maximum size in the NATCONFIG configuration module (macro NTBUFID); the buffer / work pool will then be increased only up to the specified maximum size, but not beyond. If there is no more space in the thread, variable buffers are allocated temporarily outside of the thread. The total amount of storage allocated outside the thread can be limited by profile parameter OVSZ for variable buffers and/or by profile parameter WPSZ for work pools.
- At present, the source area, the fast locate table, the PF-key table and the loop table are located in the DATSIZE buffer. With Natural Version 4.1, each of them is located in a buffer of its own.
- With Natural Version 3.1, the Natural thread is located either entirely above or below the 16 MB line. With Natural Version 4.1, a "mixed mode" is possible, that is, some parts of the thread can be above 16 MB, and at the same time other parts can be below it.
- For the handling of large alphanumeric and binary variables, Natural's thread compression has been enhanced.
- Under OS/390 in batch mode and TSO, it is possible for local buffer pools to be shared by multiple sessions.

See also Natural Storage Management in the Natural Operations for Mainframes documentation.

## Natural Global Buffer Pool

The Natural global buffer pool design and operation has changed versus previous Natural versions. To still allow processing of Natural global buffer pools from different Natural versions from one OS/390 APF authorized library, the name of the Natural Version 4.1 global buffer pool manager module is NATGBP41, and all required modules are linked to it, i.e. no other modules are dynamically loaded.

### Global Buffer Pool Changes

New commands such as ADDCACHE to add a buffer pool cache to an existing global buffer pool or DELCACHE to delete a global buffer pool's cache without terminating the buffer pool itself.

Also some commands now allow you to process global buffer pools generically by specifying a SUBSID or a BPNAM of ' \*'.

## NATSREX2/3 Example

The delivered examples NATEREX2 and NATSREX3 show how you can establish your own collating sequence for a SORT. NATSREX2 translates all bytes of a sort record into the required collating sequence, whereas NATSREX3 - after the records have been sorted - reverses this translation to the original state, see NATSREX2 and NATSREX3 - User Exits for Sort Processing in the Natural Operations for Mainframes documentation.

## Reduction of Object Code

The following measures have been taken to reduce the size of the object code generated for a Natural programming object:

- With Natural Version 3.1, entries are generated in the Variable Description Table (VDT) for every parameter, context variable and application-independent variable (AIV) defined, regardless if this field is used (referenced) in the programming object or not.  
With Natural Version 4.1, a VDT entry is generated only for those parameters, context variables and application-independent variables (AIVs) which are actually referenced in a Natural statement.  
Since Natural Version 3.1, no VDT entry is generated for local variables which are not actually referenced.
- Whenever the compiler detects an assignment of an initial value, it will generate a RESET statement instead, which will require less code. For example, for the statement MOVE 0 TO #NUMERIC, the code generated is the same as for the statement RESET #NUMERIC.
- For constants, even if they are used several times in the same program, only one entry is generated in the constant table (KST).
- The description elements for arrays will be reused for different array fields of the same structure. This reduces the object size, especially when long winded index groups are used.

The reduced object-code size of programming objects will yield several benefits:

- less storage space will be needed for the system files,
- the I/O rate for reading requested objects from the system file will be reduced,
- the Natural buffer pool can be kept smaller, etc.

## Change/Enhancement Requests

The following is an overview of the change/enhancement requests that have been implemented in Natural Version 4.1:

Request Number	Description
CE0512	Natural Advanced Facilities: new scan function for reports. (See Report Display.)
CE3291	Natural Security: allow special link of administrator to unprotected library. (See Linking Administrators to an Unprotected Library.)
CE4787	SYSBPM: the length of the field "Use Count" for the global buffer pool has been increased from 5 to 8 bytes to be able to reflect use counts exceeding 99999.
CE4866	Natural Security: new option to copy a library profile with or without links. (See Copying Libraries.)
CE5064	Natural Security: allow definition of default profiles for libraries. (See Default Profiles.)
CE5084	Natural Security: after occurrence of Adabas Response Code 9 within the current Natural session, OPRB value as determined by Natural Security is used for new database open processing.
CE5098	Support use of Julian date format in MASK option. (See MASK Option in Logical Condition).
CE5107	Natural Advanced Facilities: asterisk notation for report selection criteria. (See Report Display.)
CE5108	Natural Advanced Facilities: new selection options for deleting reports from the spool file. (See New Functions).
CE6270	Natural Advanced Facilities: new report status WAIT to trigger printing of reports not yet printed. (See Report Display.)
CE7138	SYSTP: display Natural user sessions of a specific user/terminal. (See SYSTP Utility.)
CE7808	Make Natural version number available as system variable. (See New System Variables.)
CE7892	SYSBPM: make the direct commands DISPLAY BUFFERPOOL, SELECT BUFFERPOOL and RESET BUFFERPOOL available as functions that can be selected from one of the SYSBPM menus.
CE7959	New ESCAPE statement option for leaving inline subroutine <i>and</i> programming object containing same. (See ESCAPE Statement.)
CE8586	CATALL: provide a user exit to change the default settings of the main CATALL screen. (See CATALL system command).
CE8817	NATLOAD: provide a user exit to be invoked after processing of NATLOAD has finished. (See NATUNLD/NATLOAD Utility.)
CE8886	Natural Security: the number of unsuccessful logon attempts will be passed as parameter to user exit LOGONEX1. (See Logon Procedure.)
CE8906	Roll server: User exit NATRSUX1 has been renamed to NATRSUX14 and provides additional information on each roll file (number of slots defined and in use).
CE8907	Support of IBM's LE (language environment) user-defined error handlers.
CE8944	Suppress error NAT1074 (global buffer pool not found) if an alternative buffer pool is used instead.
CE9048	Natural Security: New function to automatically update/delete the functional security of all command processors whose status is "modified". (See Functional Security.)



Request Number	Description
CE9063	Natural Security: New function to create and maintain default profiles for libraries. (See Default Profiles.)
CE9159	SYSTP: Simultaneous cancellation of multiple Natural user sessions depending on their last-activity date. (See SYSTP Utility).
CE9266	Retrieve information from the stack without using an INPUT statement (See Stack).
CE9292	Natural Security: new option to adjust the physical content of a library on the FUSER system file accordingly when a Natural Security library profile is renamed, copied or deleted. (See Copying, Renaming and Deleting Libraries).
CE9344	Natural Security: control the display of mailboxes in batch mode by a general option in Administrator Services. (See Mailboxes in Batch Mode.)
CE9451	READ statement: evaluation of ENDING AT values not done by Natural, but by the database system. (See READ Statement).
CE9452	Enhanced possibilities for usage of notation "&n&" for dynamic insertion of values in copycodes.
CE9687	Natural Security: a warning message "your password will expire in <i>nnn</i> days" will be issued to the user. (See Logon Procedure.)
P225029	Natural Advanced Facilities: enhanced PSPPATPR parameter.
P225950	INPUT statement: modifiable fields with edit masks. (See INPUT statement).
P228451	LT=0 and infinite loop needed. (See Enhanced Profile Parameters, LT).
P230559	Parameter format/length conflict not detected with PCHECK=ON. (See COMPOPT system command).
P225919	Update SYSPARM in Batch. (See Direct Commands and Batch Processing in SYSPARM Utility).

## Natural Web Interface

The following changes/enhancements to the Natural Web Interface are provided with Natural Version 4.1:

- Internal Usage of Dynamic Variables and New Natural Features
- W3\* Programs for Dynamic Variables and Dynamic Arrays of Dynamic Variables
- Use #HTTP\_NEWLINE together with W3HTML
- Samples Library SYSWEB

### Internal Usage of Dynamic Variables and New Natural Features

- Internal usage of dynamic variables instead of (A1/1:v arrays).
- Existing programs need not be recataloged.
- Use new Translate commands internally.
- The predefined maximum return page is now set to 100000 (can be changed more easily).
- To change the maximum size of the return page, only the PDA W3LIMITS needs to be changed. It is no longer necessary to change w3context, too. There is no need to change the definition of the (A1/1:v) array and its redefinition up to the maximum size that might be needed even if the majority of pages is smaller.
- Using dynamic variables decreases the data consumption of the application.
- Easier handling of environment variables greater then (A250). These variables need not be read in parts using redefined long alpha with (A1/1:v) array.
- Use of long constant values. A complete HTML page now can be defined within an init clause of the program, or an simple assign statement.

### W3\* Programs for Dynamic Variables and Dynamic Arrays of Dynamic Variables

- All interfaces of programs that had been defined as (A250) By Value have been changed to (A) Dynamic By Value. User programs need not be changed or recataloged to work with the changed interface.
- New W3\*DYNAMIC calls have been introduced. The existing calls look for trailing blanks (because of static length strings) and remove trailing blanks. The new calls take the compile dynamic variable up to \*length(), without additional checking.
- Use of constant values greater the (A253).

## Use #HTTP\_NEWLINE together with W3HTML

If more than one line should be produced and converted to HTML, it had been necessary to call W3HTMMLINE for every page. The changed Web Interface now can handle lines that had been compressed with ##HTTP\_NEWLINE.

**old:**

```
perform w3text "<pre>"
perform w3htmlline "if a > b then"
perform w3htmlline " write 'result is > then old' "
perform w3htmlline "else"
perform w3htmlline " write 'result is < then old' "
perform w3htmlline "end-if"
perform w3text "</pre>"
```

**new:**

```
perform w3text "<pre>"
COMPRESS "if a > b then" ##HTTP_NEWLINE
" write 'result is > then old' " ##HTTP_NEWLINE
"else" ##HTTP_NEWLINE
" write 'result is < then old' " ##HTTP_NEWLINE
"end-if" ##HTTP_NEWLINE
into W3DYN LEAVING NO
perform w3html w3dyn
perform w3text "</pre>"
```

## Samples Library SYSWEB

The contents of the samples directory SYSWEB have been moved to the following directory:

FNAT/SYSWEB/RES

# New Features in NOC Version 4.1

The following enhancements and corrections are provided with Version 4.1 of Natural Optimizer Compiler.

In the remainder of this document, the Natural Optimizer Compiler is also referred to as NOC.

**Note:**

Natural objects cataloged with the NOC Version 2.3 can be executed under Natural Version 4.1. or above.

- Code Generation and Performance Improvements
  - New Options
  - Revised Options
  - Incompatibilities
  - NOC 2.3 Zaps and Equivalent Option Settings
  - NOCSTAT
  - NOCSHOW
  - Option Limit for NTOPT and OPT
-

## Code Generation and Performance Improvements

With NOC Version 4.1, more compact code is generated by simplifying instruction sequences that were used for alignment-sensitive hardware.

In addition, more efficient code is generated when constants are involved, for example, `MOVE 'A' to #A(A1)` results in `MVI #A,C'A'`.

Listed below is other NOC functionality that improves code generation and program performance:

- Variable Caching
- EXAMINE
- RESET of Arrays
- Multiplication and Division of (Un)Packed Variables
- INDX Option with Parameter Arrays
- Objects with Copycode

### Variable Caching

A technique for caching variables has been implemented to speed up program execution. See Variable Caching in the section Performance Considerations in the Natural Optimizer Compiler documentation.

### EXAMINE

The EXAMINE statement has been optimized for certain clauses. See What is Compiled and What is Not in the section Using the Optimizer Compiler (Natural Optimizer Compiler documentation).

### RESET of Arrays

The code generated for executing the RESET statement for arrays has been improved.

### Multiplication and Division of (Un)Packed Variables

With NOC Version 4.1, the generated code for multiplying and dividing packed and unpacked numeric variables is more efficient.

### INDX Option with Parameter Arrays

With NOC Version 4.1, statements accessing a parameter array with a variable number of occurrences (1:V notation) are optimized even if the INDX option is specified. With NOC Version 2.3, statements accessing such a parameter array are not optimized if the INDX option is specified.

### Objects with Copycode

For Natural programming objects that contain copycode inserted with an INCLUDE statement, code generation, and hence performance, has improved significantly. This improvement is of special benefit for programming objects that contain many small or nested copycodes.

## New Options

Listed below are additional options that NOC provides to give more control over how code is generated:

Option	Function
CACHE	Switches variable caching on or off.
DIGTCHCK	Forces digits to be checked on packed and numeric fields.
SIGNCHCK	Forces signs to be checked on packed and numeric fields.
TRBASES	Puts a trace of base register allocation into the output generated by the PGEN option.
TRCACHE	Puts a trace of the cache into output generated by the PGEN option.
TRETRY	For internal use by Software AG only.

See also the section Optimizer Options in the Natural Optimizer Compiler documentation.

## Revised Options

Listed below are the NOC options that have been modified and now have a different effect under NOC Version 4.1:

- NODBG
- NOSGNTR=ON
- NOSGNTR=OFF
- LOOPS

### NODBG

The NODBG option is now also effective when INDEX, RANGE or OVFLW options are specified. This results in less code generation, so that programming objects compiled with these options execute faster. See also the section Performance Considerations in the Natural Optimizer Compiler documentation.

## NOSGNTR=ON

This option only applies to packed numeric fields of Format P. Under NOC Version 2.3, this option also applies to unpacked numeric fields of Format N.

## NOSGNTR=OFF

The COMPOPT compiler option PSIGNF=OFF is applied at compile time. Therefore, the effect of the NOSGNTR option under NOC Version 4.1 is not the same as under NOC Version 2.3, causing different signs if NOSGNTR=OFF depending on the settings of the PSIGNF compiler option. The example below illustrates the difference:

```

DEFINE DATA LOCAL
1  N(N3)
1  P(P3)
END-DEFINE
*
OPTIONS MCG=ON
RESET N P
ADD 1 TO N
ADD 1 TO P
WRITE N(EM=HHH) P(EM=HH)
*
OPTIONS MCG=NOSGNTR
RESET N P
ADD 1 TO N
ADD 1 TO P
WRITE N(EM=HHH) P(EM=HH)
*
END

```

### Output:

	PSIGNF=ON	PSIGNF=OFF
NOC23 SM1 and SM2	F0F0F1 001F F0F0C1 001C	F0F0F1 001F F0F0F1 001C
NOC23 SM3 and above	F0F0F1 001F F0F0F1 001C	F0F0F1 001F F0F0F1 001C
NOC41	F0F0F1 001F F0F0F1 001C	F0F0F1 001F F0F0F1 001C

See also Influence of other Natural Parameters in the section Optimizer Options in the Natural Optimizer Compiler documentation.

## LOOPS

This option is no longer relevant and only provided for compatibility reasons.

## Incompatibilities

Below is a list of incompatibilities between NOC Version 2.3 and NOC Version 4.1:

- NOSGNTR=ON
- NOSGNTR=OFF
- Conversion from Floating Point to Packed
- Multiplication and Division of (Un)Packed Variables
- RESET of Packed Numeric Variables
- MOVE of Numeric to Numeric with same Precision and Length

### NOSGNTR=ON

Only applies to packed numeric fields of Format P. Under NOC Version 2.3, this option also applies to unpacked numeric fields of Format N.

### NOSGNTR=OFF

The COMPOPT compiler option PSIGNF=OFF is applied at compile time. Therefore, the effect of the NOSGNTR option under NOC Version 4.1 is not the same as under NOC Version 2.3, causing different signs if NOSGNTR=OFF depending on the settings of the PSIGNF compiler option. See also Revised Options above.

### Conversion from Floating Point to Packed

With NOC Version 4.1, converting from floating point to packed numeric has been adapted to the standard Natural arithmetic, thus guaranteeing identical program results regardless of whether the NOC is used for program generation or not. When recataloged under Natural Version 4.1, for programs cataloged under NOC Version 2.3, the result can be slightly different in the last fractional parts, where relevant.

### Multiplication and Division of (Un)Packed Variables

With NOC Version 4.1, multiplication and division of packed and unpacked numeric variables have been adapted to the standard Natural arithmetic, thus guaranteeing identical program results regardless of whether the NOC is used for program generation or not. When recataloged under Natural Version 4.1, for programs cataloged under NOC Version 2.3, the result can be slightly different in the last fractional parts, where relevant.

### RESET of Packed Numeric Variables

Under NOC Version 4.1, the impact of the RESET statement on the field sign (positive) depends on the setting of the PSIGNF compiler option: If PSIGNF=OFF, the sign reads X'C'. If PSIGNF=ON, the sign reads X'F'. PSIGN=ON is the default.

Under NOC Version 2.3, the RESET statement always results in the sign X'F', unless Special Purpose Zap NA45043 has been applied (see also NOC 2.3 Zaps and equivalent Option Settings below). However, the results generated under NOC Version 4.1 are compatible with standard Natural runtime.

### MOVE Numeric to Numeric with same Precision and Length

When moving a numeric unpacked variable to another numeric unpacked variable with same length and precision, the COMPOPT compiler option NMOVE22 (see the Natural system command COMPOPT) is honored even when DIGTCHCK=ON.



## NOC 2.3 Zaps and Equivalent Option Settings

Listed below are the Special Purpose Zaps supplied for NOC Version 2.3. If none of these Zaps have been applied in your Natural Version 3.1 environment, no adaption of your NOC option setting is required. However, to migrate to NOC Version 4.1, for each Special Purpose Zap applied in your Natural Version 3.1 environment, you need to set the corresponding option(s) as described in the table below.

Zaps for NOC 2.3	Equivalent Option Setting in NOC 4.1	
	COMPOPT	NOC
NA45041	PSIGNF=OFF	NOSGNTR=OFF (default)
NA45043	PSIGNF=OFF	NOSGNTR=OFF (default)
NO31006	PSIGNF=OFF	NOSGNTR=OFF (default)
NO32003	NMOVE22=ON	
NO33005	NMOVE22=ON	
NO33007		DIGTCHCK=ON
NO34016		DIGTCHCK=ON
NO36001 NO37009 (see also below)	NMOVE22=ON	

Zap NO37009 is a correction of NO36001.

## NOCSTAT

The NOCSTAT command has been enhanced to show statistics of non-optimized programs. See NOCSTAT Command in the section Using the Optimizer Compiler.

## NOCSHOW

Minor changes have been made to the NOCSHOW utility, for example:

In batch, the whole PGEN list is output without stopping. Online, the listing is modifiable so that terminal escape codes can be entered, for example, \$\$ or << or %+.

Online, when searching, the line containing the desired text is placed at the top of the screen and the cursor is placed at the beginning of the text.

## Option Limit for NTOPT and OPT

The limit of 64 characters has been removed for option strings specified in the NTOPT macro or specified with the OPT profile parameter.

# Natural Security

The following enhancements are provided with Natural Security Version 4.1:

- Environment Protection
  - Logon Procedure
  - Concurrent Modifications of a Security Profile
  - User Profiles
  - Library Profiles
  - Utility Profiles
  - Link Functions
  - Functional Security
  - Mailboxes in Batch Mode
  - Interface Subprograms
  - Direct Commands
-

## Environment Protection

With Version 4.1, Natural Security allows you to make users' access to a library environment-specific. A Natural *environment* is determined by the combination of the system files FNAT, FUSER, FSEC and FDIC. You define a security profile for each environment (that is, for each system-file combination) you wish to protect, and control users' access to it. You can also make a library accessible in some environments, but not in others.

Whenever a user logs on to a library in another environment, Natural Security will check whether:

- access to the library is allowed in that environment, and
- the user is authorized to access that environment.

Such a check is performed not only when a user explicitly logs on to a library, but also when the user invokes a function which implicitly accesses another library or processes the contents of another library.

Environment protection is activated with a new general option "Environment Protection". When environment protection is active, the following applies:

- Access to undefined environments is not possible.
- For every environment to be accessed, an environment security profile has to be defined.
- By default, access to a library is allowed in any defined environment, and access to a defined environment is allowed for all users.
- For individual libraries and users, you can disallow access to a defined environment.

See the section Environment Profiles in the Natural Security documentation for details on environment protection.

## Logon Procedure

### Unsuccessful Logon Attempts

With Version 4.1, the number of unsuccessful logon attempts is passed as a parameter to the logon-related user exit LOGONEX1. Thus, it is possible, for example, to display corresponding information to the user *before* the maximum number of logon attempts is reached. For details, see the source of LOGONEX1.

### Password Expiration Message

With Version 4.1, you can issue a warning message "Your password will expire on *date*" (NAT1691) to users at the initial logon. The output of this message is activated with the new option "Message Before Password Expiration" in Administrator Services.

## Concurrent Modifications of a Security Profile

A new general option "Concurrent Modifications Without Notification" allows you to determine how Natural Security is to react in a situation in which two administrators simultaneously modify the same security profile. Such a situation would occur as follows:

1. Administrator 1 invokes a security profile for modification.
2. Administrator 2 invokes the same security profile for modification.
3. Administrator 1 leaves the function after having made his/her modifications - the modifications are applied to the security profile. This means that, at this point, Administrator 2 is working on data which are "out of date", but is not aware of this fact.
4. Administrator 2 leaves the function after having made his/her modifications. Depending on the setting of the new general option "Concurrent Modifications Without Notification", there are two possible reactions by Natural Security:
  - The modifications made by Administrator 2 are applied - unknowingly overwriting the modifications made by Administrator 1.
  - Administrator 2 receives a window, informing him/her that the security profile in question was in the meantime modified by another administrator. He/she can then contact the other administrator to discuss the changes made, and can then decide to either cancel his/her own modifications or apply them, thus overwriting the modifications made by Administrator 1.

With Version 4.1, this option only applies to concurrent modifications made to *mailbox* security profiles. With one of the next releases, it will also apply to concurrent modifications of other Natural Security objects.

## User Profiles

### Time Differential and Time Zone

With earlier versions, the "Time Differential" option in user security profiles is only available for users of type **TERMINAL**.

With Version 4.1, it is also available for **GROUPS**. In addition, a new option "Time Zone" is provided, which can be used as an alternative to "Time Differential".

# Library Profiles

## Private Libraries

With earlier versions, access to a private library is restricted to the user for whom the private library is defined.

With Version 4.1, it is possible to remove this restriction and control access to private libraries in the same way as access to other "normal" libraries.

With the new general option "Private Libraries in Public Mode" you can determine whether the old (Version 3) or new (Version 4.1) private-library concept is to apply: The old concept is called "private mode", the new one is called "public mode".

If you choose "private mode", the handling of private libraries remains the same as with earlier versions.

If you choose "public mode", private libraries are handled like any other libraries:

- You can choose *not* to protect a private library, which means that it can be accessed by any user.
- You can make a private library protected, which means that it can only be accessed by the user whose ID is the same as the library ID, and by users who are linked to it.

See the section Private Library in the Natural Security documentation for details.

In private mode, private libraries continue to be maintained in the user maintenance section of Natural Security; in public mode, private libraries are maintained in the library maintenance section.

## Linking Administrators to an Unprotected Library

With earlier versions, it is not possible to establish a link between a user and an unprotected library. The conditions of use of the library are determined by the library profile.

With Version 4.1, it is possible to establish a special link between a group and an unprotected library. This special link applies only to administrators contained in that group. Thus it is possible to define special conditions of use for administrators if this should be required for administration or maintenance tasks. For this purpose, the protection combination in the library profile has to be set to "People-protected=L, Terminal-protected=N".

## Copying Libraries

The Copy Library provides a new option "with links". It allows you to copy not only the library profile, but also existing links associated with that library profile.

## Copying, Renaming and Deleting Libraries

With earlier versions, when you copy, rename or delete a library security profile, this has no effect on the library itself and its contents stored on the FUSER system file.

With Version 4.1, the functions Copy Library, Rename Library and Delete Library provide a new option "with Natural objects" which allows you to also adjust the FUSER system file accordingly when a library profile is copied, renamed or deleted, which means that the contents of the library on the FUSER file are also copied to another library, moved to another library, or deleted.

## Default Profiles

With earlier versions, default security profiles can only be defined for users. With Version 4.1, you can also define default profiles for libraries.

## Statement and Command Restrictions

The use of the new statements and system commands provided with Natural Version 4.1 can also be controlled with Natural Security.

## Utility Profiles

### SYSBPM

The SYSBPM utility has been expanded to include the new functions "Write to Work File" and "Display Sorted Extract", and the new command DISPLAY CDIR; the expanded range of functions/commands is also reflected in the utility profiles for SYSBPM.

## Link Functions

With earlier versions, when you invoke a function for the maintenance of links, you would get a list of *all* objects to which the selected object can be linked, that is, those for which links already exist and those for which not.

With Version 4.1, a new option "Select only defined links" is provided; it allows you to display either a list of all linkable objects or a list of only those objects which are already linked. This new option is available for all link maintenance functions; it appears in the window displayed when you invoke a link function.

## Functional Security

If the status of a command processor is "modified" (that is, modified with SYSNCP), you have to update the functional security defined in Natural Security for the command processor. With earlier versions, you have to make this adjustment for each command processor individually. With Version 4.1, the new function "UC" of application-interface subprogram NSCLI allows you to simultaneously update the functional security of all "modified" command processors in a library.

## Mailboxes in Batch Mode

With earlier versions, mailboxes are not output in batch mode.

With Version 4.1, you can set in Administrator Services a general option "Suppress mailboxes in batch mode" which determines whether mailboxes are output in batch mode or not.

## Interface Subprograms

As of Version 4.1, "interface subprograms" are called "application interfaces".

With Version 4.1, the following new application interfaces are provided:

- NSC---P for password verification.
- NSC----P for password verification and password change.
- NSCXR14 to retrieve the user ID belonging to a specified user name.
- NSCXR15 to list all libraries of the current FNAT or FUSER system file which are not defined in Natural Security.
- NSCXR16 to translate the 2-character object-type code into the corresponding object type.

## Direct Commands

As of Version 4.1, you can also issue a Natural Security direct command from outside of the Natural Security library SYSSEC. This allows you to perform a Natural Security function from anywhere in your Natural session without having to log on to the library SYSSEC.

To do so, you enter the direct command - prefixed by "SYSSEC" - in the Natural command line.

After the function invoked by the direct command has been performed, you will be returned to the Natural screen from which you have issued the command.



# Natural SAF Security

The following enhancements are provided with Natural SAF Security Version 4.1:

- SAF Security Kernel
  - SAF Online Services
  - NSF Options
  - Documentation
- 

## SAF Security Kernel

With previous versions, the SAF Security Kernel (SAF server) was delivered as a stand-alone product. It is now part of Adabas Limited Libraries (Version 7.4.2 or above).

## SAF Online Services

The library SYSNSFOS, which contained the SAF Online Services utility, is no longer available. SAF Online Services are now contained in the library SYSSAFOS.

In Natural Security, delete any utility profiles related to the SYSNSFOS utility. To be able to use SAF Online Services, you have to define appropriate new utility profiles for the SYSSAFOS utility.

## NSF Options

When Automatic Logon is used (Natural profile parameter AUTO=ON), Natural, by default, uses the value of the Natural system variable \*INIT-USER as value for the Natural system variable \*USER. A new option "NSF \*USER Automatic Logon" (in Administrator Services, General Options 3) is provided which allows you to prevent the value of \*INIT-USER to be placed in \*USER.

## Documentation

The Natural SAF Security documentation has been revised and enhanced. In particular, a configuration example has been added.

# Natural Advanced Facilities

The following new features, changes and enhancements are provided with Version 4.1.2 of Natural Advanced Facilities:

- Spool File
  - New Functions
  - CICS-Specific Enhancements
  - IMS/TM-Specific Enhancements
  - Spool File Options
  - Server Options
  - Report Display
  - Printer Definitions
  - CALLNAT Interfaces
-

## Spool File

With Version 4.1, a new spool file layout is provided for improved data storage.

A conversion program for existing Version 2.3 spool files is also provided.

## New Functions

The following new functions are available:

- The application SYSPPOOL supports switching to other Version 4.1 spool files (dynamic change of FSPOOL assignment).
- It is possible to unload and load spool file objects and report data from/to work files (Work File Numbers 3 and 4) online.
- It is possible to copy objects from one spool file to another.
- It is possible to delete from the spool file only reports which are created before a specific date or which are older than a specific number of days.
- It is possible to switch the date/time format to storeclock (as stored on the spool file) or to the Natural time using the values currently active for the Natural profile parameters DD, TD and YD.
- There is a new program SPPPRINT to print reports in batch mode. This program supports more selection criteria such as user ID, date, etc.

## CICS-Specific Enhancements

In addition to the online definitions (CICS Options), it is possible to use the parameter module NAFPARM. Parameter settings of the NAFPARM module will then override parameter definitions on the spool file. If no parameter module is used, the online definitions on the spool file apply. The same will be true for parameters not set in the parameter module.

## IMS/TM-Specific Enhancements

In addition to the online definitions (IMS/TM Options), it is possible to use the parameter module NAFPARM. Parameter settings of the NAFPARM module will then override parameter definitions on the spool file. If no parameter module is used, the online definitions on the spool file will apply; the same will be true for parameters not set in the parameter module.

It is possible to use a spool server (IMS BMP) with the option WAIT FOR INPUT for different spool files, as the corresponding FSPOOL value is passed dynamically.

## Spool File Options

It is possible to pre-define the sequence in which reports are displayed:

- ascending by job number;
- ascending by: a) destination/form, b) status, c) priority, d) date of creation;
- ascending by: a) user ID, b) destination/form, c) status, d) priority, e) data of creation;
- automatically depending on the specified search criteria.

## Server Options

It is possible to automatically start reports whose status is WAIT (see also Report Display below).

## Report Display

- A new line command DY allows you to delete reports depending on the date or the number of days.
- A new line command SC allows you to search for texts within a report.
- The display of report details has been enhanced.
- Two new report status values are provided:
  - WAIT = Report could not be started, because the printer was active for another destination/form;
  - WOER = Error during processing of report data.
- Asterisk notation is provided for the report selection criteria.

## Printer Definitions

Control of the first/last page advance has been enhanced, allowing you to prevent a page advance being replaced by a line advance.

With earlier versions, the user exit USPSE01 in the spool server applied to all printers. With Version 4.1, you are able to use a different user exit for each printer.

## CALLNAT Interfaces

The existing interface subprograms have been expanded to cover the new functions (deletion by date/days, unloading/loading of spool file objects/report data).

# Natural Connection Version 4.1

The following Natural Version 4.1 features and restrictions apply to Version 4.1 of Natural Connection:

- Support of Long and Dynamic Variables
  - Work File Format and Attributes
  - Installation Procedure
  - Maximum File Transfer Record Length for Natural Connection
-

## Support of Long and Dynamic Variables

With Natural Version 4.1, the sizes allowed for alphanumeric and binary variables have been increased as described in Size of Alphanumeric and Binary Variables in the section Programming Language Enhancements. In addition, you can allocate the length of such variables dynamically at runtime as described in Dynamic Variables in the section Programming Language Enhancements. These new Natural Version 4.1 features can be applied when uploading or downloading Natural Connection work files. Prerequisites for the use of the new features are:

- Natural Connection Version 4.1
- Entire Connection Version 4.3.2
- If installed: Entire Screen Builder Version 5.1.1 PL 1

## Work File Format and Attributes

Below are the restrictions that apply to the use of work file attributes:

- Accessing PC work files is restricted to a fixed record length of 1073741823 bytes or 32767 bytes when using the statement `WRITE WORK FILE VARIABLE`. Depending on the Entire Connection version installed on the PC, additional restrictions may apply as described below.
- Natural Connection does not support work files of the type `UNFORMATTED`.  
A work file is always transferred in formatted mode and contains record-oriented data only. For a work file of the type `UNFORMATTED`, at `FILE OPEN`, Natural Connection switches to the type `FORMATTED`. To transfer byte-streamed data, see Streaming in the Natural Connection documentation.

## Installation Procedure

The installation procedure for Natural Connection has been revised completely. As the NTCPC utility has been dropped, the `INPL` file and the corresponding installation step have been omitted.

## Maximum File Transfer Record Length for Natural Connection

As of Natural Connection Version 4.1, the maximum record length supported for file transfer is 1 GB - 1 byte = 1073741823 bytes. For more information, see Maximum File Transfer Record Length for Natural Connection in the Natural Connection documentation.

# Natural RPC Version 5.1.1

The Natural Remote Procedure Call is available as a separate subcomponent of Natural and has its own version. The version of the Remote Procedure Call is oriented at the version of Natural on Windows and UNIX. This measure takes into account that the Natural Remote Procedure Call is a cross-platform component and makes it possible to provide new Natural RPC versions independent of new Natural versions for the various platforms supported.

Part of the RPC functionality provided with Natural Version 4.1 for Mainframes had already been delivered with Natural Version 3.1.6 as optional features.

The following new product features, changes and enhancements are provided with Version 5.1.1 of Natural Remote Procedure Call.

## New Features:

- Support of Large and Dynamic Alpha and Binary Variables
- Maximum Length for Node and Server Names Increased to 32 Characters
- Support of the EntireX Broker ACI V6
- Support of SSL for the TCP/IP Communication
- Support of EntireX Location Transparency
- New Application Programming Interface USR2035N
- New Application Programming Interface USR2073N
- New User Exit NATRPC99
- Support of Natural Security Logon without Valid Password
- Enhanced Verification of the Natural Security User ID
- Prerequisites

## Enhanced Features:

- Support of Multiple EntireX Broker Logons (USR2071N)
- Enhancements to Application Programming Interfaces
- Enhancements to Status Function RPCINFO

## SYSRPC Utility Enhancements and Modifications:

- Generation of Service Directory (NATCLTGS) in User Library
- Support of Long Node and Server Names
- Support of EntireX Location Transparency
- New Properties for Local Directory
- Generation of Stubs in User Library
- Removal of 32K Restriction for Stubs
- Preserving Attribute Definitions of Existing Stubs
- Attribute Definitions as Comment in DEFINE DATA PARAMETER
- Support of Natural Security Utility Profiles

## Changed Features:

- Implicit END TRANSACTION in a Conversation
- Release of Adabas Retain Sets
- Reworked Sample USR1071P
- Changes to RPCERR
- Changes to the Error Messages of Server
- Changes to PIng and TErminate Messages

- Service Directory (NATCLTGS) in User Library
- Generation of Stubs in User Library
- Changes/Enhancements to Profile Parameter RPC
- RPCSIZE Considerations

**Unsupported Features:**

- Optional Parameters
-



## New Features

- Support of Large and Dynamic Alpha and Binary Variables
- Maximum Length for Node and Server Names Increased to 32 Characters
- Support of the EntireX Broker ACI V6
- Support of SSL for the TCP/IP Communication
- Support of EntireX Location Transparency
- New Application Programming Interface USR2035N
- New Application Programming Interface USR2073N
- New User Exit NATRPC99
- Support of Natural Security Logon without Valid Password
- Enhanced Verification of the Natural Security User ID
- Prerequisites

### Support of Large and Dynamic Alpha and Binary Variables

With Natural RPC Version 5.1.1, large alpha/binary and dynamic alpha/binary formats are supported in the parameter list of a remote CALLNAT execution. In case of dynamic alpha or binary variables, the remote CALLNAT on the server may increase or decrease the size of the dynamic variables received. Only the size currently used is sent back to the client.

Dynamic alpha and binary variables are only supported with automatic RPC execution, that is without using Natural RPC stubs.

In the case of dynamic alpha and binary variables, the client uses the value of the Natural RPC profile parameter MAXBUFF for the receive buffer length passed to the EntireX Broker stubs. If Entire Net-work is used as transport layer and the Natural profile parameter ACIVERS is set to 3 or above, the value for MAXBUFF must not exceed 30.

#### **Note:**

Large and dynamic variables must only be used by a Natural RPC client if they are also supported by the corresponding RPC server. If large and dynamic variables are sent to an RPC server that does not support them, the RPC server will reject the request.

### Maximum Length for Node and Server Names Increased to 32 Characters

With Natural RPC Version 5.1.1, the maximum length for node and server name has been increased to 32 characters to be compliant with the EntireX Broker ACI. This enhancement allows you to specify a fully qualified TCP/IP node name and makes the etc/hosts and etc/services definitions obsolete.

Neither the interface nor the internal structure of the local directory NATCLTGS has been changed. See also Support Long Node and Server Names below.

### Support of the EntireX Broker ACI V6

The Natural RPC profile parameter ACIVERS has been enhanced to enable you to specify Version 6.

The specification of ACI Version 6 is strongly recommended if you are using the TCP/IP stub EXAAPSC (CICS only) to enable Natural to use the TERMINATE option for the LOGOFF from the EntireX Broker.

#### **Note:**

The EntireX Broker stub in use must support the API version specified with ACIVERS. For the supported API versions, refer to the EntireX documentation.

## Support of SSL for the TCP/IP Communication

Secure Socket Layer (SSL) support for the TCP/IP communication to the EntireX Broker has been introduced. To enable the EntireX Broker stub to recognize that the TCP/IP communication should use SSL, you have to use one of the following methods:

- Append the string :SSL to the node name.
- Prefix the node name with the string SSL://

To use SSL, an SSL parameter string must be passed to the EntireX Broker stub on the very first call. For this purpose, the new application programming interface USR2035N has been provided.

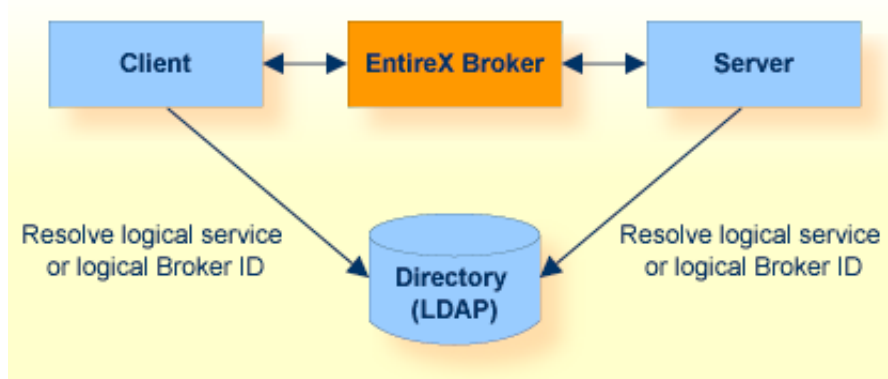
For more details about SSL and the SSL parameter string, see the EntireX documentation.

**Note:**

The EntireX broker stub EXAAPSB or EXAAPSC must be used for this feature.

## Support of EntireX Location Transparency

With EntireX, location transparency is possible. Instead of using the physical node name and the physical server name, a server can be addressed by a logical name. This makes the location of the EntireX Broker and the name of the server transparent to clients and servers. The logical name is mapped to the physical node and server names by the EntireX Broker stub before it is used the first time.



For more details about the EntireX Location Transparency, see the EntireX documentation.

To take advantage of location transparency, the Natural RPC has been enabled to accept a logical name wherever only a physical node and a physical server name could be specified before.

The maximum length of a logical name is 192 characters. To avoid new Natural profile parameters, a logical name is specified in the server name and node name part of the existing parameters. There are two kinds of logical names:

- **Logical node names**

With a logical node name, you specify a logical name for a node only in conjunction with a physical server name.

- **Logical services**

With a logical service, you specify a logical name for both the node and the server. To define a logical service, the node name has to be set to \*, and the server name contains the logical service name.

The following components refer to node and server names:

- The keyword parameters SRVNODE, SRVNAME, DFS and RDS of the NTRPC macro for static specification, or the subparameters with the same name of the profile parameter RPC.
- Service maintenance of the SYSRPC utility
- Service directory (NATCLTGS)
- Application Programming Interfaces USR2007N, USR2071N
- Service programs RPCERR, RPCINFO

The new information about logical service names is stored in the local directory NATCLTGS without changing its interface or its internal structure. All information is stored as attribute/ value pairs and the logical service names are just added under a new attribute.

To support existing callers, the interface to USR2007N and USR20071N has not been changed. To be able to retrieve or specify long logical service names, the respective internal PDA fields have been defined with the VALUE RESULT option and their length has been increased.

**Note:**

The EntireX broker stub EXAAPSB or EXAAPSC must be used for this feature. In addition a directory service is required.

## **New Application Programming Interface USR2035N**

For the support of the Secure Socket Layer (SSL) communication, the new application programming interface USR2035N is provided to set the required SSL parameter string. USR2035N must be called before the first remote procedure call is executed on the client side, or before the Natural RPC server logs on to the EntireX Broker.

## **New Application Programming Interface USR2073N**

For the support of pinging and terminating a Natural RPC server from within a Natural program, the new application programming interface USR2073N is provided.

## New User Exit NATRPC99

An optional user exit named NATRPC99 has been provided to support customer-specific termination logic for a Natural RPC server. NATRPC99 is called after the Natural RPC server has deregistered and logged off from the server node. If no NATRPC99 is found, the server terminates immediately as usual. If a Natural program with name NATRPC99 is found, the server continues to run as a normal Natural session and NATRPC99 is called with a FETCH statement.

## Support of Natural Security Logon without Valid Password

The interface between the Natural RPC server and Natural Security (NSC) has been enhanced to support a client logon without a valid password. The client has to generate the logon data as before but the password must not be a valid NSC password for the user. This feature is similar to the automatic logon (AUTO=ON) for local Natural sessions.

For details see the NSC documentation about the Natural RPC session parameters.

## Enhanced Verification of the Natural Security User ID

The interface between the Natural RPC server and Natural Security (NSC) has been enhanced to support the verification of the Natural user ID of the client versus the EntireX user ID used by the client to logon to the EntireX broker. This enhanced verification is especially useful if the NSC logon is done without a valid password (see above) and the EntireX Broker logon has been validated by a security system.

For details, see the NSC documentation about the Natural RPC session parameters.

## Prerequisites

- EntireX Broker stubs EXAAPSB/EXAAPSC (which require SMARTS) if you want to use SSL and/or location transparency.
- Directory services if you want to use location transparency.

For details on the EntireX Broker stubs and their SMARTS requirement, see the EntireX documentation.

## Enhanced Features

- Support of Multiple EntireX Broker Logons (USR2071N)
- Enhancements to Application Programming Interfaces
- Enhancements to Status Function RPCINFO

### Support of Multiple EntireX Broker Logons (USR2071N)

The application programming interface USR2071N has been enhanced to allow you to log on to multiple EntireX Brokers concurrently. That is, if you have already issued a logon to an EntireX Broker, a logon to a new EntireX Broker no longer implies a logoff from the current one.

### Enhancements to Application Programming Interfaces

To support long node and server names, the application programming interfaces USR2007N and USR2071N have been enhanced to accept and return node and server names having a length of up to 192 characters. Existing callers who are using 8-character-long names will still work and need not be adapted.

### Enhancements to Status Function RPCINFO

To support long node and server names, the RPCINFO subprogram has been enhanced to return the up to 32 character long physical node and server names. Existing callers, which use 8 character long names will still work and need not be adapted. For compatibility reasons, the RPCINFOL local data area still uses 8-character-long node and server names.

## SYSRPC Utility Enhancements and Modifications

With Natural RPC Version 5.1.1, the following changes and enhancements have been made to the Natural SYSRPC utility:

- Generation of Service Directory (NATCLTGS) in User Library
- Support of Long Node and Server Names
- Support of EntireX Location Transparency
- New Properties for Local Directory
- Generation of Stubs in User Library
- Removal of 32K Restriction for Stubs
- Preserving Attribute Definitions of Existing Stubs
- Attribute Definitions as Comment in DEFINE DATA PARAMETER
- Support of Natural Security Utility Profiles

A short description of these changes and enhancements is given below. For more details, refer to the SYSRPC utility documentation.

### Generation of Service Directory (NATCLTGS) in User Library

The generated service directory (subprogram NATCLTGS) is stored in the current user library. For this reason, you are recommended to LOGON to the application library (or one of its STEPLIBs) used by the client at runtime before you invoke the SYSRPC utility.

### Support of Long Node and Server Names

To be compliant with the EntireX Broker, the Service Directory Maintenance function enables you to specify node and server names of up to 32 characters. The editing functions of the Service Directory Maintenance function have been adapted accordingly.

### Support of EntireX Location Transparency

To support the EntireX Location Transparency, the Service Directory Maintenance function enables you to specify logical node names and logical services of up to 192 characters. The editing functions of the Service Directory Maintenance function have been adapted accordingly.

### New Properties for Local Directory

The property "transport protocol" has been added, which makes the Natural profile parameter ACIPATT obsolete.

The RDS specific properties "expiration time" and "unique directory identifier" are integrated into the local directory using appropriate attributes.

### Generation of Stubs in User Library

The generated stubs are stored in the current user library. For this reason, you are recommended to LOGON to the application library (or one of its STEPLIBs) used by the client at runtime before you invoke the SYSRPC utility.

## Removal of 32K Restriction for Stubs

The length restriction of 32 KB for the data exchanged between client and server has been removed. It is now possible to generate stubs, which send/receive up to 1 GB of data.

In addition, the stub generation will display the required length for the send and receive buffer. This will help the administrator to adjust the EntireX Broker attribute file definitions that are related to the message length.

For information about the EntireX Broker attribute file definitions, see the EntireX documentation.

## Preserving Attribute Definitions of Existing Stubs

If an existing stub is regenerated (e.g. to add or delete parameters), the Attribute fields on the stub generation screen are preset with the attribute definitions that were assigned to the parameters when the stub was last generated. Therefore, you are no longer required to re-enter the attribute definitions.

## Attribute Definitions as Comment in DEFINE DATA PARAMETER

For better readability, the attribute definitions that have been assigned to the parameters on the stub generation screen are added as comment in the DEFINE DATA PARAMETER area.

## Support of Natural Security Utility Profiles

The use of the various functions of the SYSRPC utility can be controlled by Natural Security utility profiles. For information about protecting utilities, see the Natural Security documentation.

## Changed Features

- Implicit END TRANSACTION in a Conversation
- Release of Adabas Retain Sets
- Reworked Sample USR1071P
- Changes to RPCERR
- Changes to the Error Messages of the Server
- Changes to PIng and TErminate Messages
- Service Directory (NATCLTGS) in User Library
- Generation of Stubs in User Library
- Changes/Enhancements to Profile Parameter RPC
- RPCSIZE Considerations

### Implicit END TRANSACTION in a Conversation

If ETEOP=ON had been set on the Natural RPC server side in previous Natural RPC versions, an implicit END TRANSACTION statement had been issued at the end of the execution of each remote subprogram. This may have lead to inconsistent data in the database if a conversation had been established which should be rolled back as part of the database transaction, but the modifications were already committed.

With Natural RPC Version 5.1.1, ETEOP=ON has no effect on the conversation, that is, no implicit END TRANSACTION is issued inside a conversation.

An implicit END TRANSACTION is still executed after execution of the last remote CALLNAT of a conversation, if ETEOP=ON is specified.

### Release of Adabas Retain Sets

At the end of a non-conversation CALLNAT and at the end of a conversation, a RELEASE SETS is issued to release all Adabas retain sets. This ensures that the next request (which may be for a different client) will not see the data.

### Reworked Sample USR1071P

The prime usage of the application programming interface USR1071N is to set a user ID and password that is passed to the RPC server with a following RPC request. The reworked sample program USR1071P on the library SYSEXT reflects this fact and only accepts user ID and password. The original version of USR1071P has been renamed to USR1071X. It provides extended functionality that in addition enables the setting and/or retrieval of various data.

### Changes to RPCERR

To support long node and server names, the RPCERR program shows the up to 32 character long physical node and server names. The display window has been adapted accordingly.

### Changes to the Error Messages of the Server

Error messages that are detected by the Natural RPC server before the remote CALLNAT is executed and that are directly caused by the remote CALLNAT (e.g. security violations, incompatible data format), are reported with the program name of the remote CALLNAT and the virtual line number 9999.



## Changes to Ping and Terminate Messages

The Ping and Terminate messages have been enhanced and indicate that there is a Natural RPC server (in contrast to an EntireX RPC server) and the operating system where the Natural RPC server is running.

## Service Directory (NATCLTGS) in User Library

The service directory (subprogram NATCLTGS) is no longer generated into the library SYRPC but into the current user library. If you want to generate NATCLTGS in the library SYSRPC, you must first LOGON to the library SYSRPC before invoking the SYSRPC utility.

### Note:

It is strongly recommended that you generate NATCLTGS in the user library and leave the library SYSRPC on the FNAT unchanged.

## Generation of Stubs in User Library

The stubs are no longer generated into the library SYRPC but into the current user library. If you want to generate a stub in the library SYSRPC, you must first LOGON to the library SYSRPC before invoking the SYSRPC utility.

### Note:

It is strongly recommended that you generate the stubs in the user library and leave the library SYSRPC on the FNAT unchanged.

## Changes/Enhancements to Profile Parameter RPC

The following changes/enhancements have been made to the Natural profile parameter RPC:

The following RPC subparameter has been dropped:

<b>RPC Subparameter</b>	<b>Purpose/Reason</b>
ACIPATT	Define Node Pattern for ACI Protocol This subparameter has become obsolete since ACI is the only supported transport method.

The following RPC subparameters have been enhanced:

<b>RPC Subparameter</b>	<b>Enhancement</b>
DFS	The range of possible values for <i>server name</i> and <i>server node</i> has been extended to 192 characters.
MAXBUFF	The range of possible values has been extended to 2097147 KB.
RPCSIZE	The range of possible values has been extended to 2097151 KB.
SRVNAME	The range of possible values has been extended to 192 characters.
SRVNODE	The range of possible values has been extended to 192 characters.

## **RPCSIZE Considerations**

The new Natural RPC requires about 1 KB more working storage in the (client and server) settings of the RPCSIZE profile parameter that determines the size of the buffer used by the Natural RPC.

Depending on your environment, you may have to increase the RPCSIZE accordingly.

## **Unsupported Features**

### **Optional Parameters**

Optional parameters (*nX* notation in the CALLNAT statement) are not yet supported in the parameter list of a remote CALLNAT execution. This support will be provided with the forthcoming Natural RPC 6.1 version.